IBM

SC21-7803-1
S5280-28

IBM 5280 Distributed Data System

Introduction to DE/RPG

SC21-7803-1
S5280-28

IBM 5280 Distributed Data System

Introduction to DE/RPG

Second Edition (May 1980)

This is a major revision of, and obsoletes, SC21-7803-0. Changes or additions to the text and illustrations are indicated by a vertical line to the left of the change or addition, except Chapters 5 and 6. Chapters 5 and 6 have been revised extensively to support changes to the Source Entry program. You should read these two chapters in their entirety.

This edition applies to release 1, modification 0 of the IBM 5280 DE/RPG (Program 5708-DE1), and to all subsequent releases and modifications until otherwise indicated in new editions or technical newsletters.

Changes are periodically made to the information herein; these changes will be reported in technical newsletters or in new editions of this publication.

Use this publication only for the purposes stated in the *Preface*.

This publication contains examples of data and reports used in daily business operations. To illustrate them as completely as possible, the examples include the names of individuals, companies, brands, and products. All of these names are fictitious and any similarity to the names and addresses used by an actual business enterprise is entirely coincidental.

It is possible that this material may contain reference to, or information about, IBM products (machines and programs), programming, or services that are not announced in your country. Such references or information must not be construed to mean that IBM intends to announce such IBM products, programming, or services in your country.

Publications are not stocked at the address below. Requests for copies of IBM publications and for technical information about the system should be made to your IBM representative or to the branch office serving your locality.

This publication could contain technical inaccuracies or typographical errors. Use the Reader's Comment Form at the back of this publication to make comments about this publication. If the form has been removed, address your comments to IBM Corporation, Product Information Development, Department 997, 11400 Burnet Road, Austin, Texas, 78758. IBM may use and distribute any of the information you supply in any way it believes appropriate without incurring any obligation whatever. You may, of course, continue to use the information you supply.

Preface

This manual is intended to be used by individuals who want to learn how to use the data-entry functions provided by the IBM 5280 DE/RPG (Data Entry with RPG subroutines) Program Product. Upon completing this manual, the reader should be able to create the simple data-entry jobs needed for his data-entry business applications and should understand the organization of DE/RPG well enough to understand the techniques and applications that will be described in the *DE/RPG User's Guide*.

Chapters 1 through 6 contain the information necessary to create simple data-entry programs. This type of program satisfies the requirements of most data-entry environments. Chapters 7, 8, and 9 describe advanced data-entry applications that involve programming concepts such as the use of tables.

The first two chapters of this manual describe general data-entry and DE/RPG terms and concepts. The experienced person might want to read these chapters for a brief review. The inexperienced person should carefully read Chapters 1 and 2. Both readers should review the information on the back of the general utility (Z) and data description (A) specifications to determine the type of functions that are available. They may also want to scan the *DE/RPG Reference Manual*, which contains detailed descriptions of the functions provided by DE/RPG.

Related Publications

- *IBM 5280 General Information* manual, GA21-9350, describes the devices and program products available with the 5280 system.
- *IBM 5280 Operator's Guide*, (to be available at a later date), provides a description of the processes involved in operating the 5280 system.
- *IBM 5280 Planning and Site Preparation Guide*, GA21-9351, provides information relevant to installing the 5280.
- *IBM 5280 Utilities Reference/Operation Manual*, SC21-7788, provides information about using the noncommunications utilities.
- *IBM 5280 System Control Programming Reference/Operation Manual*, GC21-7824, provides detailed information about the SCP for the 5280.
- *IBM 5280 DE/RPG User's Guide*, (to be available at a later date), provides tips and techniques for programmers using DE/RPG.
- *IBM 5280 DE/RPG Reference Manual*, SC21-7787, provides detailed information about DE/RPG.

Contents

Chapter 1. About this Manual

This manual teaches you how to use DE/RPG (Data Entry with RPG subroutines) to write programs for your data-entry jobs. This involves teaching you DE/RPG terminology and concepts so you can use the *DE/RPG Reference Manual*. This manual contains only the basic DE/RPG functions, the reference manual describes all DE/RPG functions. The *DE/RPG User's Guide* contains descriptions and sample programs for the more complex operations that DE/RPG can perform.

Chapter 2 gives an introductory description of data-entry concepts and terms. This description does not contain everything you need to know about data entry. It does, however, contain the information that you need to know in order to use DE/RPG for data entry.

Chapters 3, 4, 5, and 6 explain the process of defining a simple, data-entry job using DE/RPG. This process consists of:

- Reviewing the source document to determine the arrangement of the data (Chapter 3)
- Identifying the arrangement of data to use for writing the data set on the diskette (Chapter 3)
- Determining the data checks, shifts, and edits that are needed to ensure accurate entry (Chapter 3)
- Defining the appearance of the displays that will be used in the job (Chapter 3)
- Using the data description specification (A) to describe the fields on the display, the records containing the fields, and the data set where records will be stored on the diskette (Chapter 4)
- Using the general utility specification (Z) to describe the formats, their order of use, and the job (Chapter 4)
- Entering the sample program using the source entry program (Chapter 5)
- Compiling the program to prepare it for use by the operator (Chapter 5)
- Using the compiled program to enter data (Chapter 6)

When you have completed Chapter 6, you will understand how to use DE/RPG to write a program that creates a master customer file. You will also be familiar with some of the basic DE/RPG data-entry functions and you should begin to feel comfortable with the terms, specifications, and processes involved. *Remember that the more involved you are in working through the sample programs, either by making entries on your own specifications or by trying another similar program, the more you will learn.*

Chapter 7 assumes that you know how to write a simple data-entry program; it concentrates on teaching you how to use additional editing and automatic functions performed by DE/RPG. Tables are introduced and used in a variety of ways in this chapter. Again, there is only one sample program, and it takes you through the process of designing the displays for the second data-entry job.

Chapter 8 completes the job process by writing the program using the A and Z specifications.

Chapter 9 teaches you how to write a program that creates tables.

Chapter 10 contains self-test questions that you should answer to measure your understanding of DE/RPG.

Appendix A provides answers to the questions that are asked at the end of the chapters. It also contains solutions to the sample program descriptions given in Chapters 6 and 10. Answer these questions to gain the maximum use of the manual.

Appendix B contains a glossary that provides a quick reference source for definitions of unfamiliar terms.

Appendix C contains blank display work sheets and specification forms for your use.

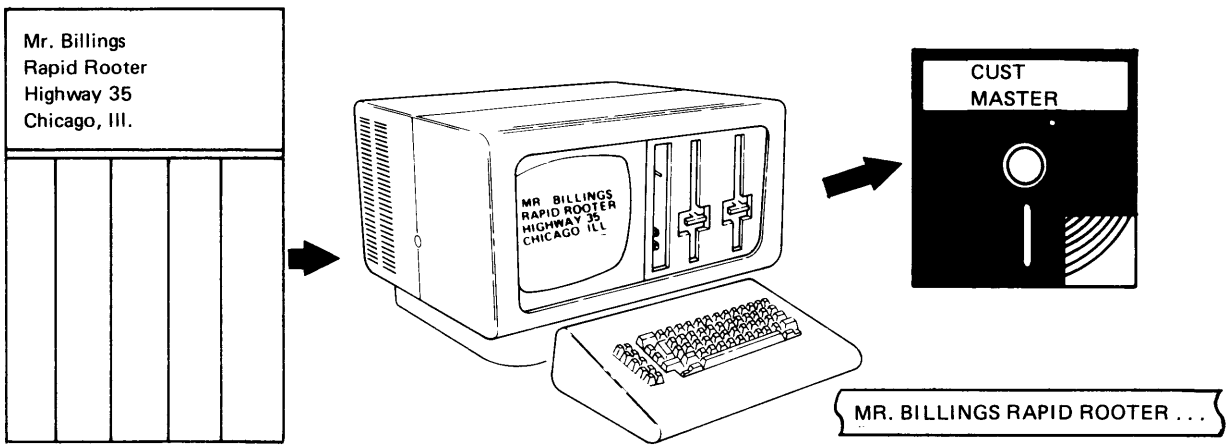
In this manual, whenever samples of display work sheets or specifications are being used, the area being described appears in color.

Chapter 2. Introduction to the Concepts of Data Entry

This chapter describes some basic (1) data-entry concepts and (2) DE/RPG terms that you will need to know.

WHAT THE TERM *DATA ENTRY* MEANS IN THE 5280 SYSTEM

Data entry is the process of transferring information from an existing source (such as an order entry form) to a diskette record.



Source Document Display Format Diskette Format

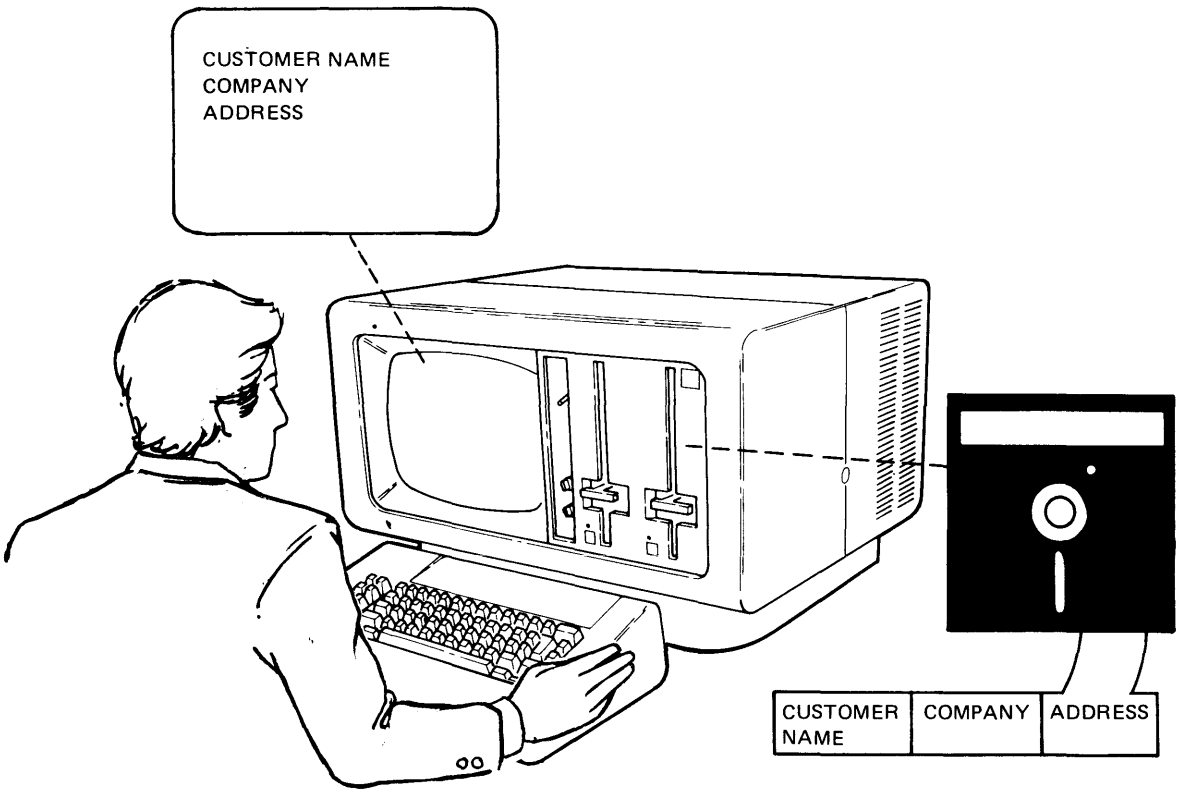
DE/RPG fulfills the requirements of data entry by providing a way for you to write programs for data-entry jobs. DE/RPG allows you to *format* information for the display and for the diskette, to perform *checks* and *edits* against the data as it is entered, and to perform *automatic functions* that reduce the number of entry keystrokes.

You should understand certain basic *data-entry concepts* and *terms* before you can begin using DE/RPG. These are:

- Job
- Formats
- Checking and editing
- Automatic functions

Job

A job defines and controls the data-entry task you want performed. For example, a data-entry task might consist of allowing someone to enter information from a source document and then write the information to a diskette. Entering information from an order form and writing the data to a diskette for use in billing or inventory is an example of how a DE/RPG job can be used.



Formats

Formats determine the sequence of information on the display and diskette; in other words, formats specify what goes first, what follows, and what is last. DE/RPG allows you to define the formats you require for the job.

There are two types of display formats: entry formats and review formats. Entry formats specify the sequence to be used for displaying a record during the enter mode. Review formats specify the sequence to be used for displaying a record when you are changing or verifying data. Modes are described later in this chapter.

Checking and Editing Functions

The next basic concepts that you should understand are *checking* and *editing*. Both terms are used to define the process of placing some limitations on the data as it is entered. DE/RPG provides checks and edits such as the following to help you ensure that the entered data is correct:

- Requiring the operator to enter something
- Right-adjusting partially filled fields and adding either zeros or blanks into the unfilled positions
- Requiring the operator to fill the field with data
- Allowing only one type of entry such as numeric only or alphabetic only

There are many more checks and edits that DE/RPG performs for you. The *DE/RPG Reference Manual* describes each of these in detail.

Automatic Functions

The final basic data-entry concept that you should know is *automatic functions*. Automatic functions are actions that can be performed by DE/RPG without the operator's intervention. DE/RPG provides a variety of these functions such as:

- Automatically duplicating one entry from another entry
- Inserting characters
- Providing messages to guide the operator
- Performing arithmetic operations such as adding, subtracting, multiplying, and dividing

HOW DE/RPG RELATES TO DATA ENTRY

So far you have learned that DE/RPG allows you to design data-entry jobs, create data-entry formats, specify checks and edits against data as it is entered, perform automatic functions, and write data in a diskette data set. To do these operations, you write a DE/RPG program using the general utility (Z) and data description (A) specifications.

Figure 1 illustrates the specifications you will learn to use to write programs in this manual.

IBM International Business Machines Corporation

IBM 5280 GENERAL UTILITY SPECIFICATIONS

Printed in U. S. A.

Job		Keying Instruction	Graphic								Description	Page of
Operator	Date		Key									

Z

Sequence	Form Type	Name Type	Format ID (1-9, A0-Z9)	Job/Format/ Subroutine Name	Reserved	Repeat (1-9, N)	Mode (E/R)	AND(A)	Test Conditions: Position to be Tested (*POSnnnn)	Reserved	Condition	Character to Test for ('C')	Reserved	Next Format ID (0-9, A0-Z9)	Reserved	Options: Job Line / Entry Lines
01	Z								*POS		EQ	' '				CFILE (data set) CLRL (number) DATE (*DMY *YMD) EOJ (*JOB *PASS) EDITC (cuptd) SLNO (line) ENTRATR (2attr) WRITE (name) EXITATR (2attr) JOBOPT ([*NOPMT][*NOOPEN]) PRTFILE (data set) SHARE (names) SHARER (names) STATUS (name) TFILE (data set [deffreq]) 2attr=BL,CS,HI,ND,RI,UL
02	Z								*POS		EQ	' '				
03	Z								*POS		EQ	' '				
04	Z								*POS		EQ	' '				
05	Z								*POS		EQ	' '				
06	Z								*POS		EQ	' '				
07	Z								*POS		EQ	' '				
08	Z								*POS		EQ	' '				
09	Z								*POS		EQ	' '				
10	Z								*POS		EQ	' '				

IBM International Business Machines Corporation

IBM 5280 DATA DESCRIPTION SPECIFICATIONS

Printed in U.S.A.

Job No.	Dataset	Keying Instruction	Graphic								Source Document	Page of
Operator	Date		Key									

A

Sequence	Form Type	Comment (*)	Reserved	Indicator (for CHECK (BY, BV) or ERROR)	Reserved	Name Type (F/R/K/T)	Reserved	Dataset/Record/ Field/Table Name	Reserved	Length	Data Type	Reserved	Decimal Positions (0-9)	Usage (I/O/B/W)	Location: Line	Location: Pos	Editing: Checks=CHECK (code . . .) / Functions
01	A																Checks: Auto Dup =AD; Auto Skip =AS; Blank Check =BC; Bypass =BY; Bypass on Verify =BV; Data Required =DR; Dup Disable =DD; Field Exit Required =FE; Lower Case =LC; Mandatory Entry =ME; Mandatory Fill =MF; Rt Adj–Blank Fill =RB; Right to Left =RL; Rt Adj–Zero Fill =RZ; Self Check =nxx; D=M/G (Check/Gen); xx=Modulus. Functions: ADD (name); AUXDUP (name); AUXST (name); COMP (1test fld1 2@ fldn 'literal' [indicator]); DSPATR (2attr); EDTCDE (code *float); ERROR (code [*message]); EXSR (subroutine); INSERT (fld1 2@ fldn 'literal'); LOOK (table [index]); PMT (prompt); RANGE (low high); RANGET (table [index]); RESET ([*TOTn] [name]); SEQ (1test); SETOF (ind); SETON (ind); SHIFT (4shift); SUB (name); SUBST (table1 table2 [index1]); TADD ([*TOTn] [name]); TSUB ([*TOTn] [name]); XCHK (table index1 index2) 1test=EQ,GE,GT,LE,LT,NE 2attr=BL,CA,CS,HI,ND,RI,UL 3@=*,-,+,/ 4shift=A,D,H,N,V,W,X,Y 'literal'
02	A																
03	A																
04	A																
05	A																
06	A																
07	A																
08	A																
09	A																
10	A																
11	A																
12	A																
13	A																
14	A																
15	A																
	A																
	A																
	A																
	A																
	A																

1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20 21 22 23 24 25 26 27 28 29 30 31 32 33 34 35 36 37 38 39 40 41 42 43 44 45 46 47 48 49 50 51 52 53 54 55 56 57 58 59 60 61 62 63 64 65 66 67 68 69 70 71 72 73 74 75 76 77 78 79 80

*Number of sheets per pad may vary slightly.

Figure 1. General Utility and Data Description Specifications

You must understand some *DE/RPG concepts* and *terms* to use DE/RPG effectively. These concepts are:

- Programs
- Files
- Data Sets
- Modes
- Records
- Fields

Programs

A DE/RPG program is the information you provide to describe the job. A minimum program must contain descriptions for the job, data set, modes, files, records, and fields. Details about these descriptions will be provided when you start writing your first sample program in Chapter 4.

Files

Files receive and temporarily store data. Files are related to input/output devices; they define the interface between DE/RPG and the 5280 devices.

The *display file* receives data from the keyboard and controls the data on the display. The *diskette file* controls the contents of the diskette data set. When a complete display record is assembled, the data in the display record is written in a diskette record described by the diskette file. Every data-entry job that is interactive (requires the operator to use the keyboard) uses programs that contain file descriptions for at least two files: display and diskette.

Data Sets

A data set is the collection of related records on the diskette. Using a DE/RPG program to enter information into the system is one way to create a data set.

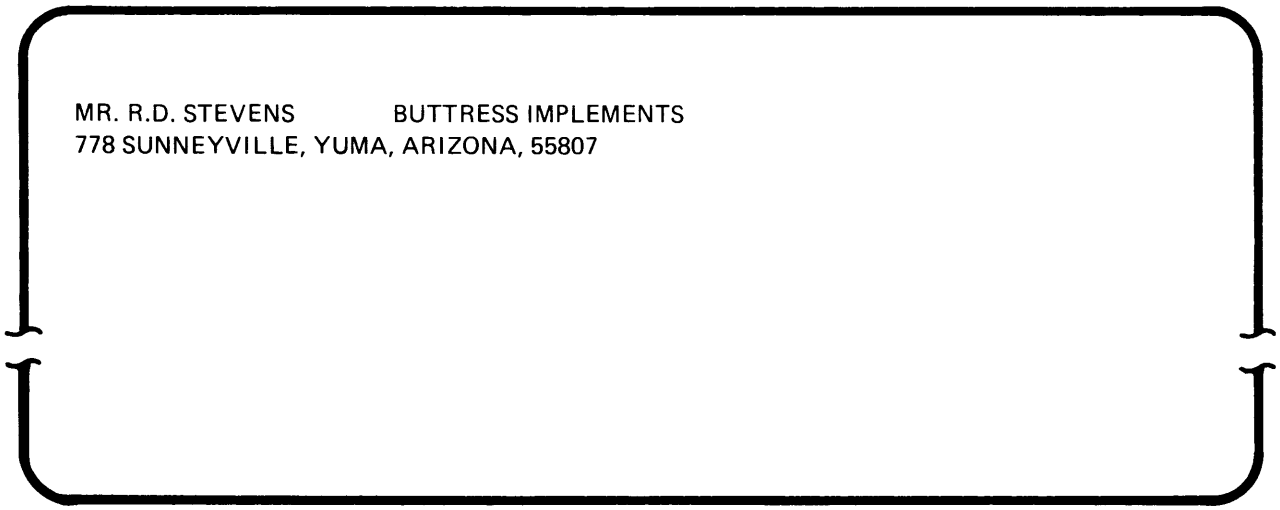
Modes

Modes are types of operations during which data entry can be performed. Four primary modes are used by DE/RPG: enter, update, verify, and rerun. Enter mode allows the operator to enter data. Update mode allows the operator to change data. Verify mode helps the operator check the accuracy of the entries made during the enter mode. Rerun mode provides accurate totals and uninterrupted automatic calculations. No operator interaction occurs during the rerun mode.

Records

A record can be thought of as a unit of related information. There are two kinds of records: records for the display and records for the diskette.

A display record is the contents of a single display. For example, a display that contains all the information needed to describe a customer can be thought of as a display record.



Display Record

The diskette record consists of data entered by the operator or data automatically supplied by DE/RPG. It need not contain all the descriptive information that was on the display to guide the entry.

This same information appears as follows for the diskette record.

MR. R.D. STEVENS BUTTRESS IMPLEMENTS778 SUNNEYVILLE, YUMA, ARIZONA, 55807

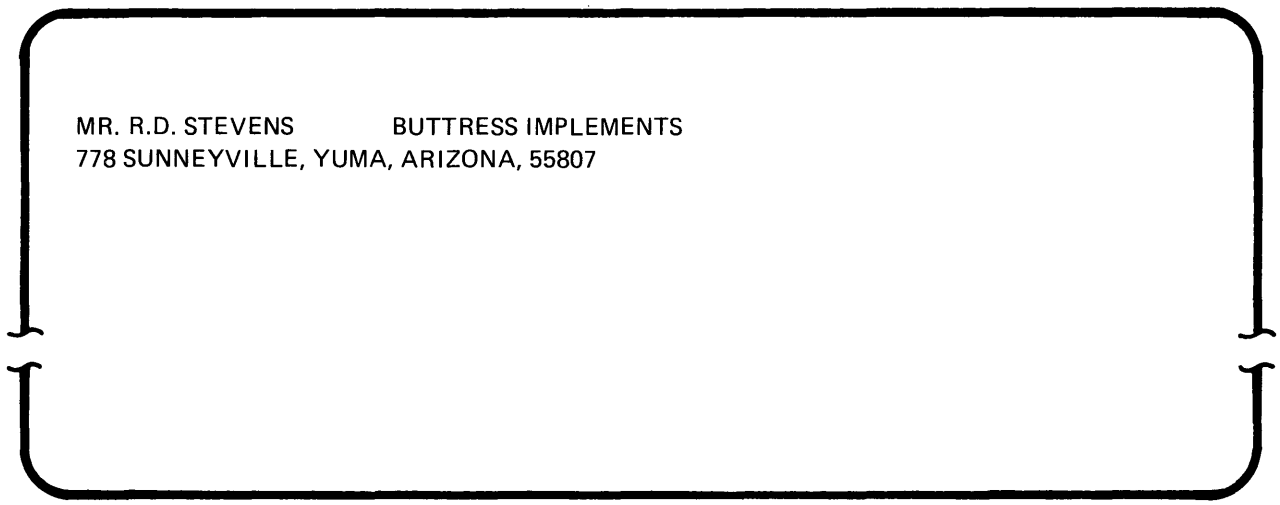
Diskette Record

Notice that there is space separating some information in the record. This separation is formed by the unused part of each field. Normally, entries are not as long as the length specified for the field. When an entry is shorter than the space allowed for it, blanks (or, in some types of data, zeros) fill the unused portion.

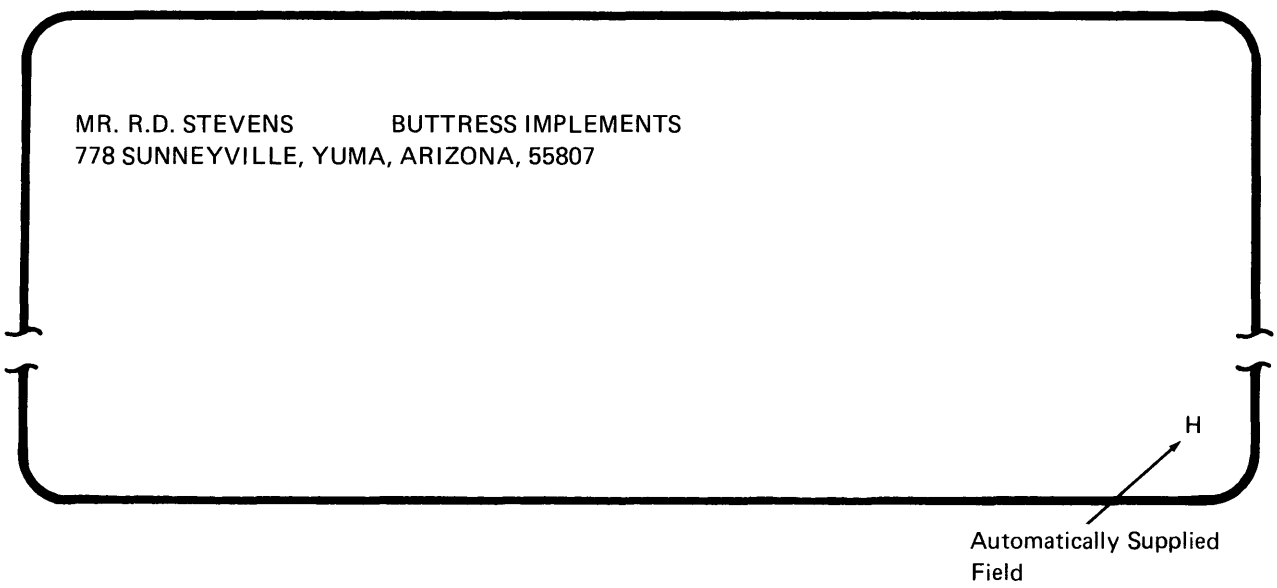
Fields

Each record consists of smaller pieces of related data called fields. A field can consist of (1) information supplied by DE/RPG or (2) data entered by the operator.

Look at the sample display record again.



The fields shown on this display (MR. R.D. STEVENS, BUTTRESS IMPLEMENTS, 778 SUNNEYVILLE, YUMA, ARIZONA, 55807) are all data fields that must be supplied by the operator. Now look at the following example to understand how a field supplied by DE/RPG might be used.



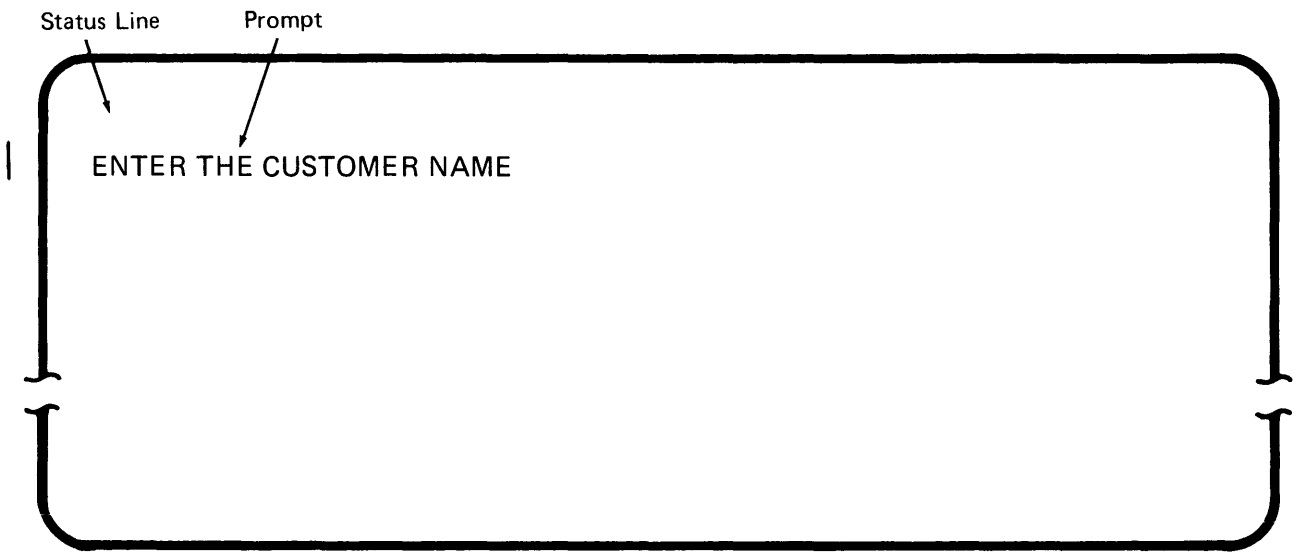
Notice that a new field has been added to the record. In the lower corner of the display, the letter H has been added. This field can be used to distinquish this type of record from other records. Because this record contains header-like information, the H can be used to mark the record as a header record. If this record were combined with other records containing detailed data, it could be distinquished from the detailed records as a header record type by this H identifier. Header information is general information such as a customer's name and address. The three basic types of data-entry records (header, detail, and trailer) are described in Chapter 7.

DE/RPG can automatically include the record markers you specify. No checks are performed against automatically supplied fields. The edits and checks are performed against the data as it is entered. Edits and checks against manually entered fields immediately identify invalid entries as errors.

Prompts and Literals

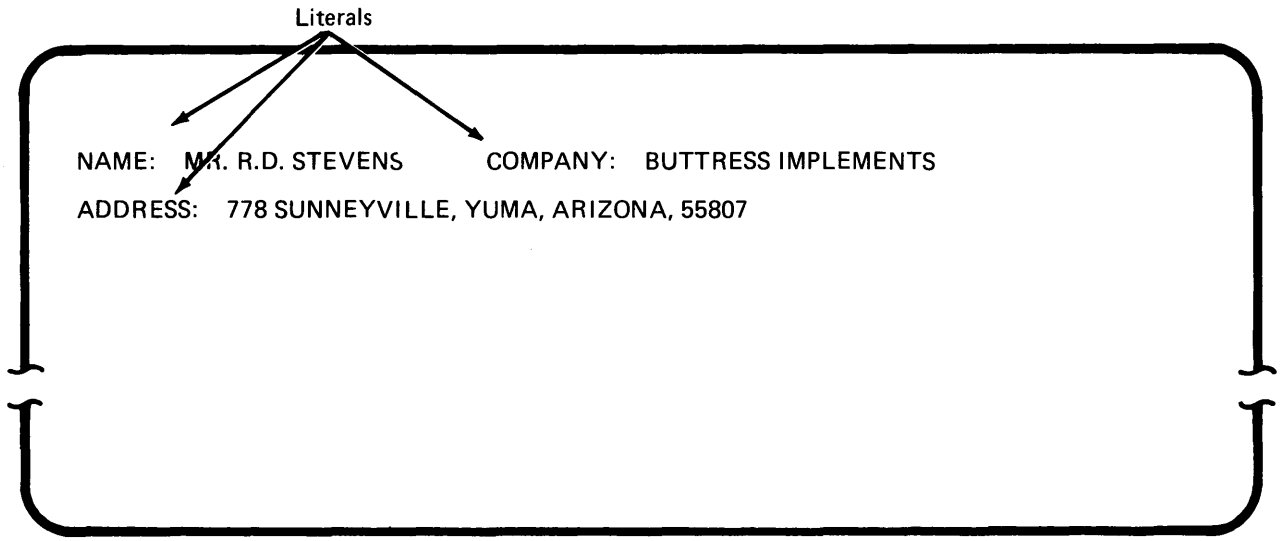
Explanatory information for describing fields or records may be specified in two ways: prompts and literals. Both are controlled by DE/RPG.

A prompt is information that is used to tell the operator what information the field requires; the prompt always appears on row 2 of the display and is displayed until the field is exited. Row 1 of the display is always used by the system for a status line.



The literal is a special kind of prompt that appears next to the field it describes or by itself. It remains on the display until the complete record is entered.

Look at the same sample display with literals added:



The literals in this example are NAME:, COMPANY:, and ADDRESS:. Neither the prompt nor the literal is written on the diskette. This means that the display contains the prompts and literals, but the diskette does not.

The contents of the diskette record for the previous example is:

MR. R.D. STEVENS BUTTRESS IMPLEMENTS 778 SUNNEYVILLE, YUMA, ARIZONA, 55807

SUMMARY OF CHAPTER 2

You should now have a basic understanding of data entry and DE/RPG concepts and terms. To test your understanding, try to answer these questions:

1. Match the following terms with their definitions:

   Terms

   1. Job _____
   2. Files _____
   3. Data set _____
   4. Format _____
   5. Modes _____
   6. Record _____
   7. Field _____
   8. Prompt _____
   9. Literal _____
   10. Program _____

Definitions

a. The information that describes the task to be performed.

b. The types of operations in which the program can operate.

c. Device-related objects that temporarily receive and store data.

d. Pieces of related information that make up the contents of the display or diskette.

e. A fixed-position message that appears on the display.

f. The task to be performed.

g. A message that can appear anywhere on the display.

h. The smallest pieces of related information that can be in the display or on the diskette.

i. The collection of related records on the diskette.

j. The organization of information on the display or on the diskette.

2. In your own words, describe what the term data entry means.

_____

_____

3. In your own words, describe what DE/RPG does.

_____

_____

Check your answers against the answers provided in Appendix A. If you have not answered these questions correctly, reread this chapter. The next chapter will start the process of defining the first sample job.

Chapter 3. Getting Started Using DE/RPG

You can use DE/RPG to define and control the format of information on the display and on the diskette, check or edit information as it is entered, perform automatic functions, and control the appearance of data on the display.

OVERVIEW OF THE PROCESS INVOLVED IN USING DE/RPG

Before you begin the first sample job, you should have an understanding of the total process involved in using DE/RPG. Figure 2 provides this overview.

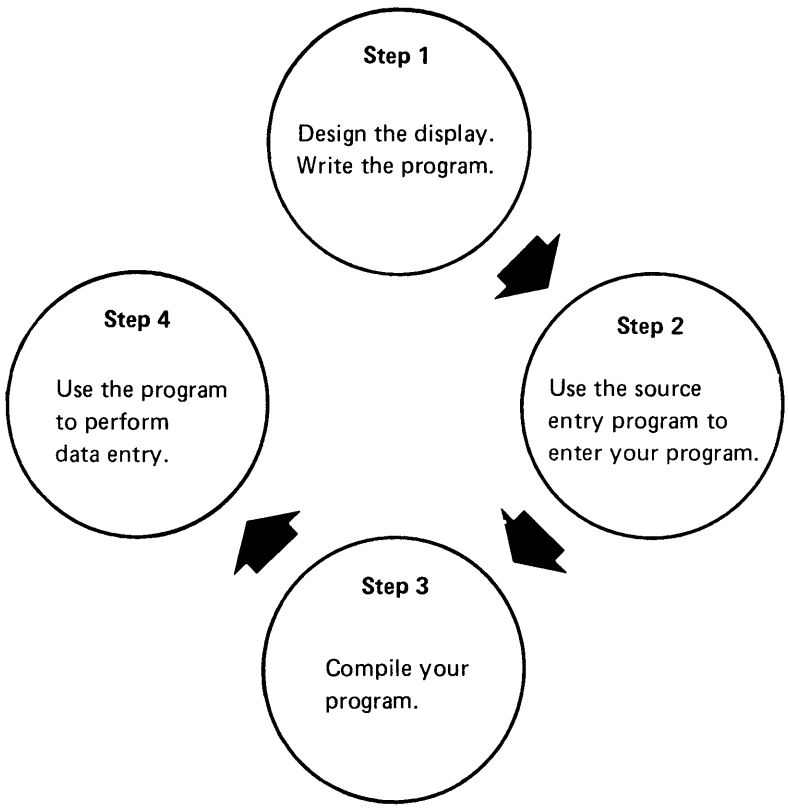


Figure 2. Overview of the Process Involved in Using DE/RPG

The source entry program mentioned in Step 2 of Figure 2 is available with DE/RPG. This program prompts you for your program information, which is on the Z and A specifications. Chapter 5 shows you how to use the source entry program.

The compiler mentioned in Step 3 of Figure 2 is supplied on the DE/RPG diskette. All programs that you write using DE/RPG must be compiled. Chapter 5 shows you how to compile the program for the first sample job.

Review Step 1 in Figure 2. This is where you are now.

In this chapter, you will start the first sample job. The first thing you will do is to design the displays for the DE/RPG program. When you have read this chapter, you should be able to do the following:

- Describe the fields in the source document that need to be defined for the program
- Design displays that look like the source document
- Assign checks and edits to the fields
- Define the format for the records in the diskette data set

Until you have carefully planned what you want the operator to see, you should not use the specifications to define the job, format, files, records, or fields. Once you have designed the displays on work sheets, describing them on the A specification is simple.

THE DESCRIPTION OF THE MASTER CUSTOMER IDENTIFICATION JOB

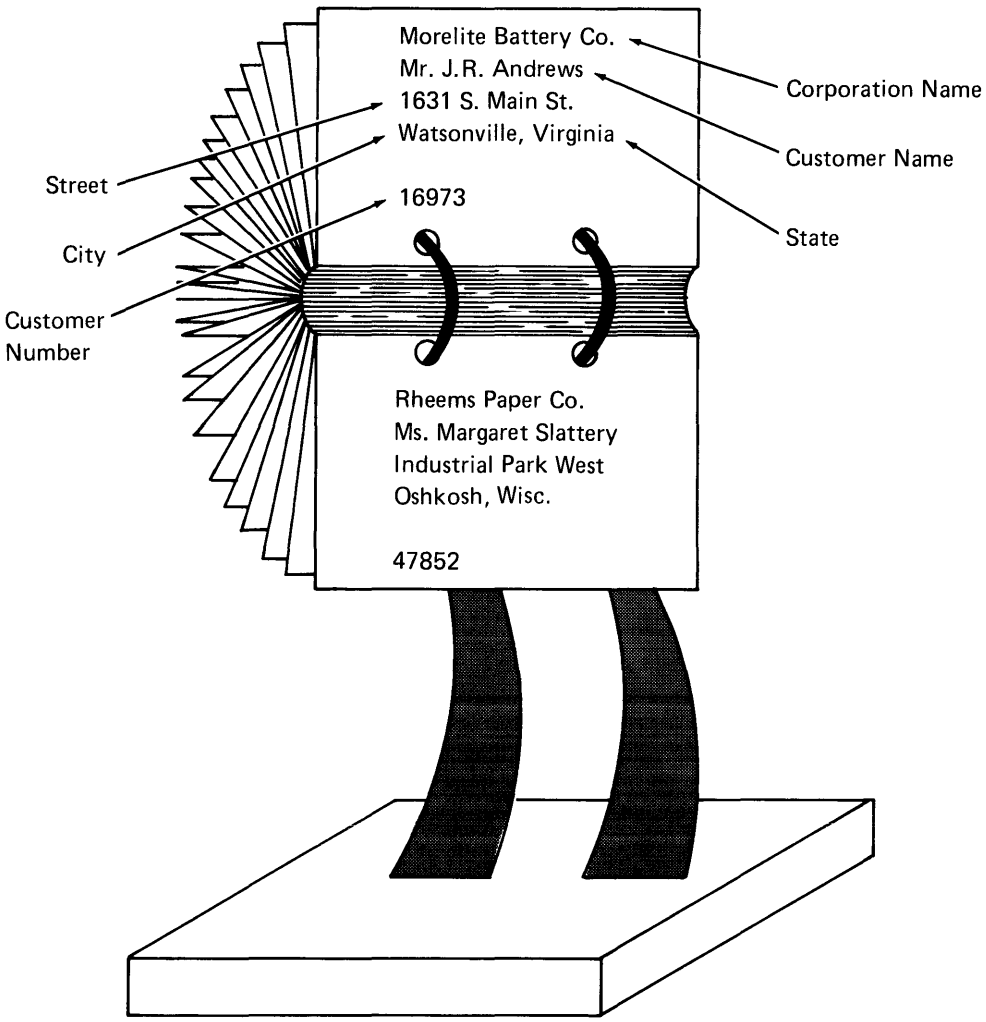
Using DE/RPG, you design the format for the display and for the diskette. Normally, you will be given the source document that the person entering data uses. This source document might consist of a salesperson's filled-in order blank or, as in this example, cards from a customer address file. You will want to design a display format that uses the source document as a model; this helps the person entering the data to do the most efficient job.

The person who uses the data set provides you with the diskette format that is required. Without this format, the information you provide might be useless. This diskette format may or may not be the same as the display format. If the formats are different, the operator is unaware that the information is being written on diskette in a format that is different from the format being displayed. In the first job, the display and diskette formats are the same.

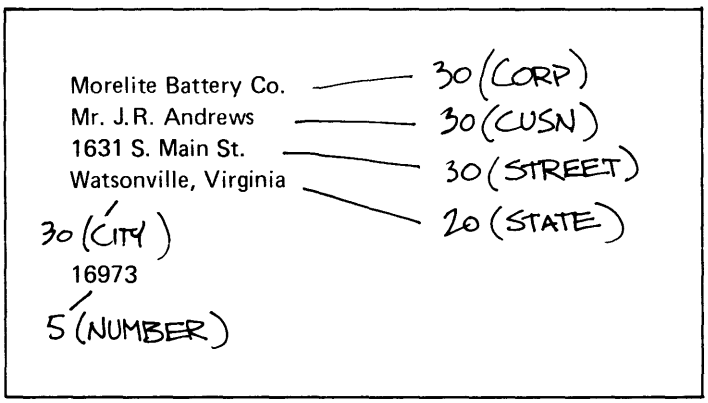
To understand how to best define a data-entry job, you must be familiar with the environment in which it will be used. Consider the data-entry department that will be creating the first sample program. This department is small; it has three people who enter data and design programs. Often they exchange entry tasks. This means that the person who designs the program might not always be entering data for it. There is a need, then, to include enough prompting to guide the operator.

The supervisor of the data-entry department has given you a customer address file. You have been instructed to write a DE/RPG program that will allow an operator to enter the information from the cards in the file and write it in a diskette data set.

This is the customer name and address card file.



The name of the data set you are to create is MASTHEAD. Assume that the supervisor has told you that the data on each card will form one record and that all records are 150 positions long. The record name is to be HEADER. In addition, you are to mark an H in the last position of each diskette record, so that all records are identified as header records and can be used later in another job. The supervisor has also taken a card from the file and written the length and names of the fields for each entry. You will use these lengths when you design the displays; you will use the names when you write the program.



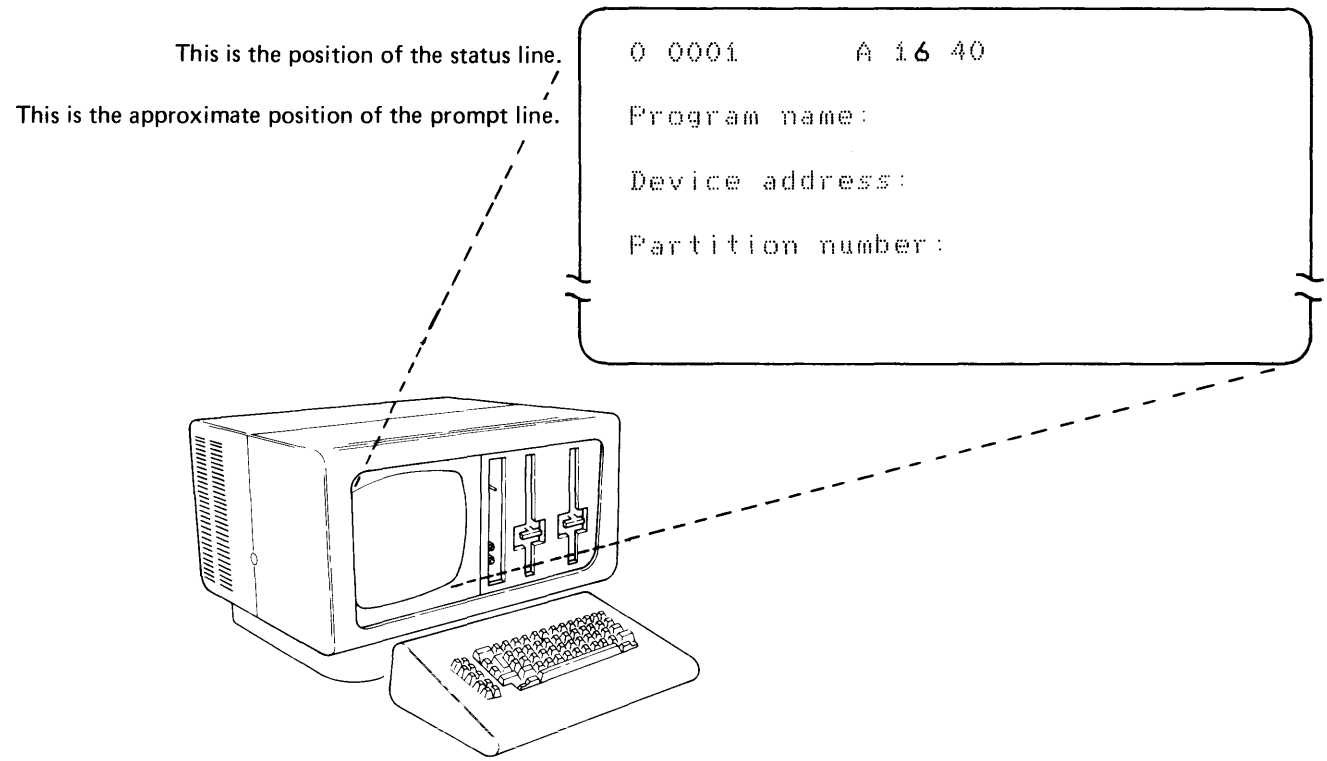
Jobname = MASTER

Place an H in the last position of the diskette record. Use the format of the card for the diskette record. 1 record per card. Name the fields as shown. Data set name is MASTHEAD. Use format 1D X1. Insert 1 deleted record after every ten records that are written. Make the records 150 positions long with an H marked in the last positon

You now have enough information to start designing the job. The remainder of this chapter and Chapters 4, 5 and 6 show you how to use DE/RPG to design this job, write the program for it, enter the program into the 5280, and use the program to write customer data on the diskette.

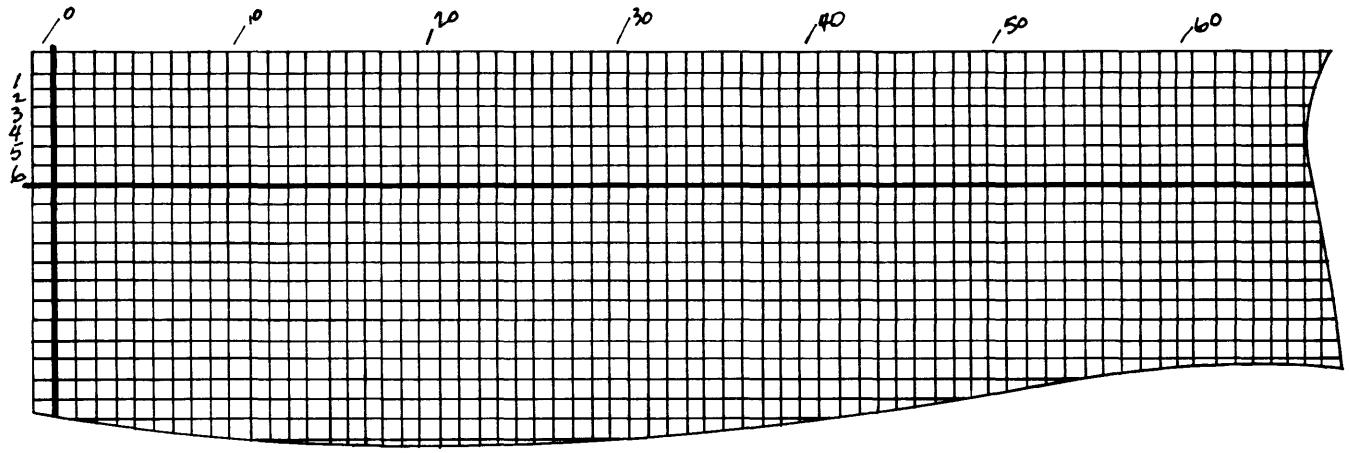
DESIGNING THE DISPLAYS

The easiest way to start is to decide on the appearance of the displays. To do this, you should use work sheets. Assume that you will be using a 480-character display. This display is 6 rows high and 80 columns wide.



You can use either marked graph paper or the display screen layout sheets, GX21-9271, as work sheets. This manual uses the latter to illustrate display designs. If you use graph paper, be sure to mark 80 columns across the top and 6 rows down the side as a guide for placing the fields on the work sheet.

Graph Paper



Display Screen Layout Sheet

Design Layout Sheet

COLUMN

1–10	11–20	21–30	31–40	41–50	51–60	61–70	71–80
1 2 3 4 5 6 7 8 9 0	1 2 3 4 5 6 7 8 9 0	1 2 3 4 5 6 7 8 9 0	1 2 3 4 5 6 7 8 9 0	1 2 3 4 5 6 7 8 9 0	1 2 3 4 5 6 7 8 9 0	1 2 3 4 5 6 7 8 9 0	1 2 3 4 5 6 7 8 9 0

The method used for marking the display work sheet is a matter of individual choice. The important part of marking the work sheet is that field descriptions are clear and correct.

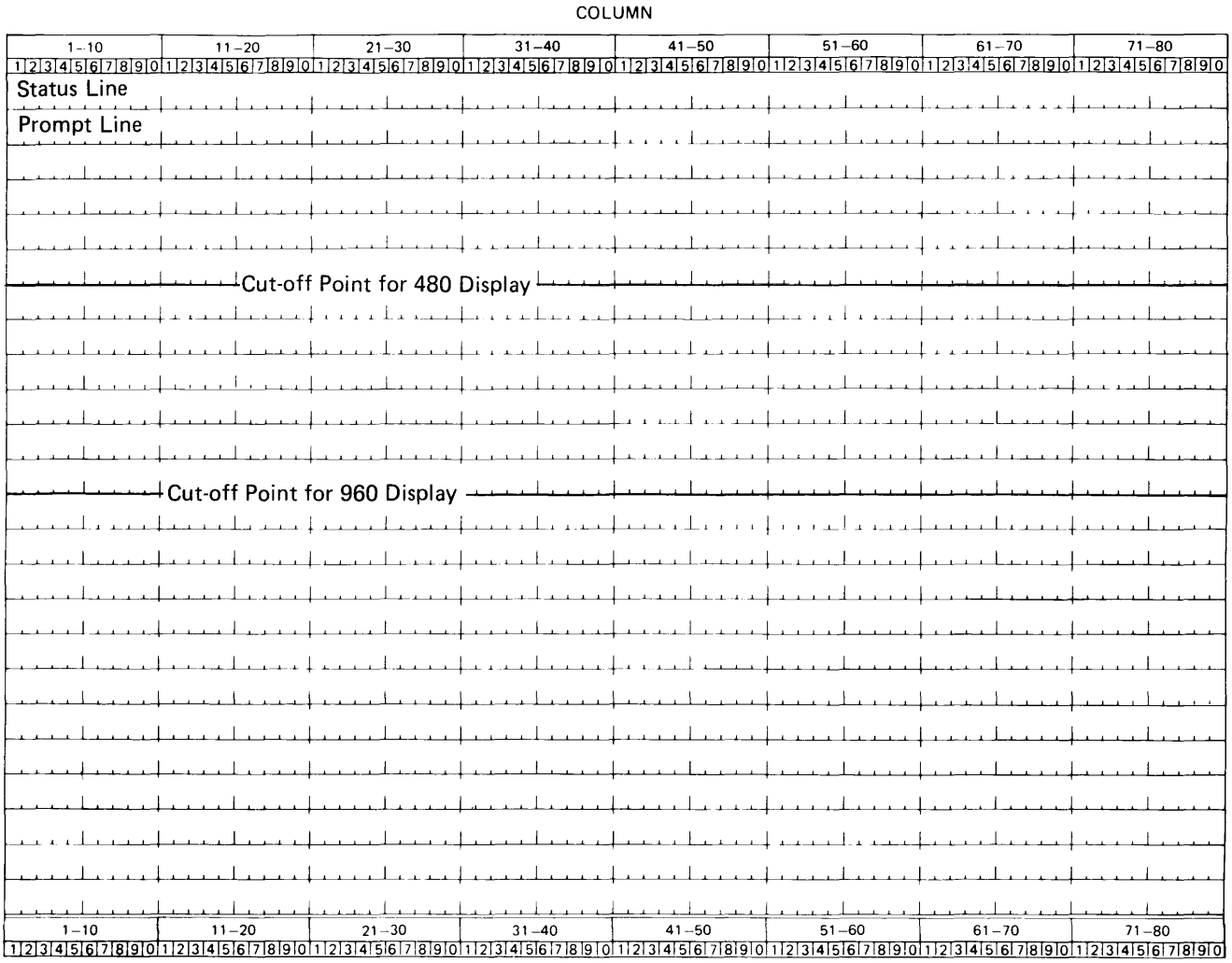
Factors to Consider When Marking the Work Sheets

Before you begin to mark the display work sheets, you should understand how DE/RPG uses the display. The first row of each display is not available for your general use. This row is reserved for status information; the status line is described in Chapter 6. Generally, the status information tells you such things as the mode in which the program is operating, the record number currently being displayed, error conditions, and the number of positions remaining in the field.

The second row of every display is used for fixed-position prompts included in the DE/RPG program. DE/RPG places all prompts in column 1 and row 2 of the display.

A prompt can be a maximum of 200 characters long. Prompts are always associated with a data field. When the data for the field has been supplied and the next field is displayed, the prompt disappears. The row used by the prompts can also be used for any other field defined by the program; if fields and prompts both occupy this row, the prompt overlays all fields on the row.

Display Screen Layout Sheet



When you use DE/RPG, the second row of the display (one down from the status line) is considered line 1, the next row is line 2, and so on. In the sample programs, you will learn how to reserve row 2 for prompts and use rows 3 through 6 for data fields and literals.

The line and column information for the display is described on the A specification. When you use the A specification, you provide the line (row) and column position for each field you want formatted on the display. If you do not want to position fields on the display, do not include this information. When the line and column information is not included for the program, the first field begins in row 3 and all remaining fields are strung together following the first field. All prompts appear in row 2.

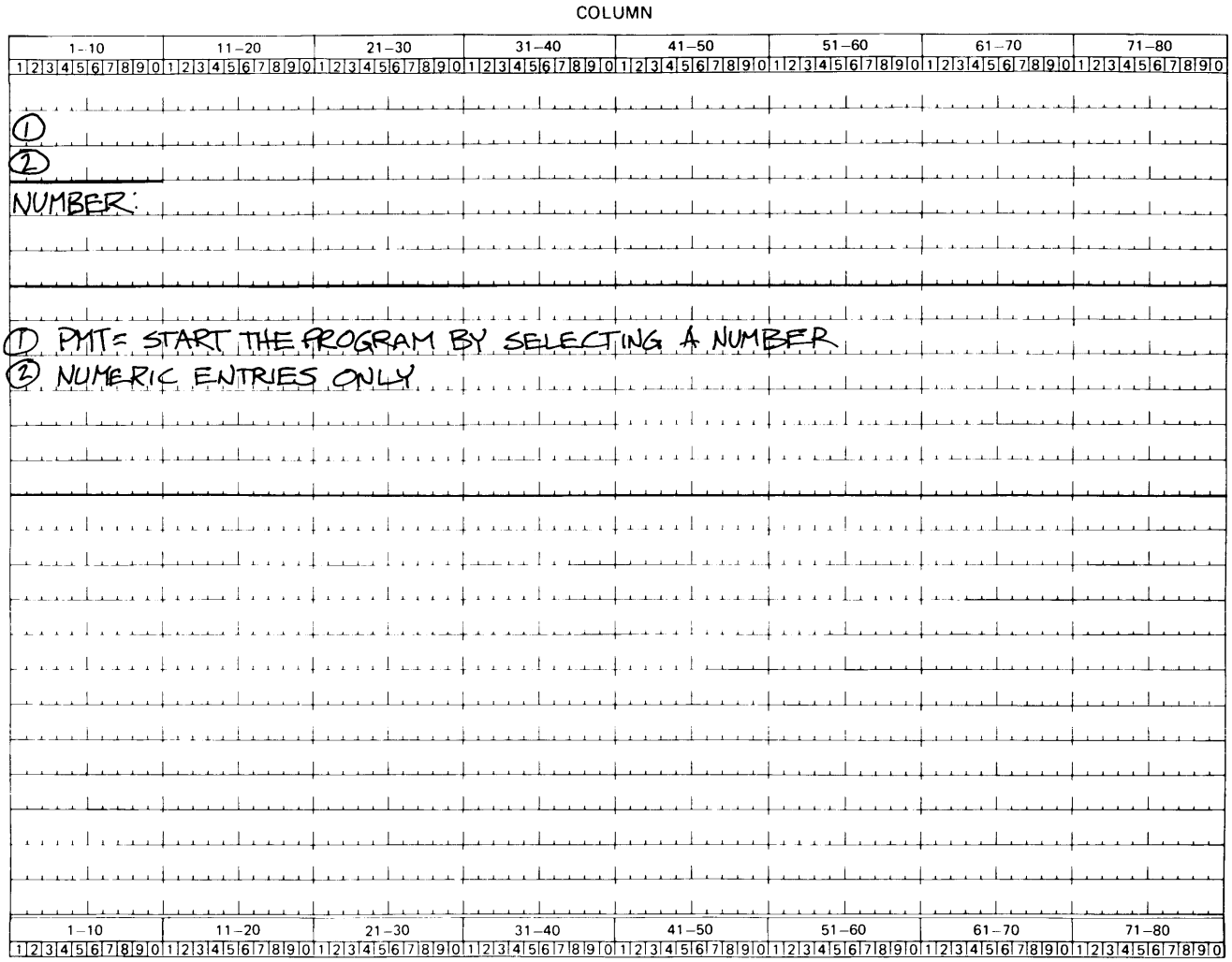
When you want to format fields on the display, all prompts appear in row 2. The program considers *row* 2 to be the *first line*. Fields described as being in line 1 and column 1 begin in row 2 and column 1, where they are overlayed by prompts. If you do not want fields overlayed by prompts, you need to specify that they begin in line 2 and column 1 or below.

Both sample programs in this manual use formatted displays with prompts using row 2 (line 1) and data fields and literals beginning in row 3 (line 2).

A Brief Description Showing How the Work Sheets are Marked

The following sample indicates the way the work sheets will be marked.

Display Screen Layout Sheet



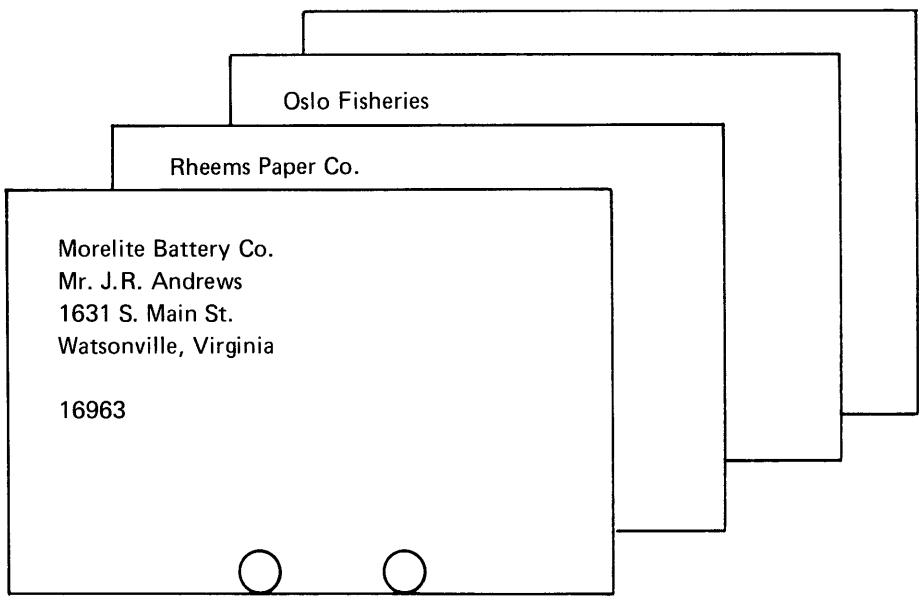
Notice that line 0 is blank. Line 1 contains a circled number which is repeated in the lower part of the work sheet. This circled number is a key that corresponds to the line on the bottom of the work sheet with the description of the field. The field being described is a prompt. This prompt will always appear on line 1 (the location of the first circled number). The wording for the prompt is START THE PROGRAM BY SELECTING A NUMBER.

Line 2 of the work sheet contains a straight line that goes from column 1 to column 10. It also has a circled number. The line is indicating that this is a data field. The length of the line tells you how long the field should be. The circled number references a description of the field. In this case, the description tells you that the field should only contain numeric entries. Line 3 contains the word NUMBER:. The word NUMBER names the literal to be placed in this location of the display.

Now that you know how the displays will be marked on the work sheets, you are ready to start describing the fields for the first sample job.

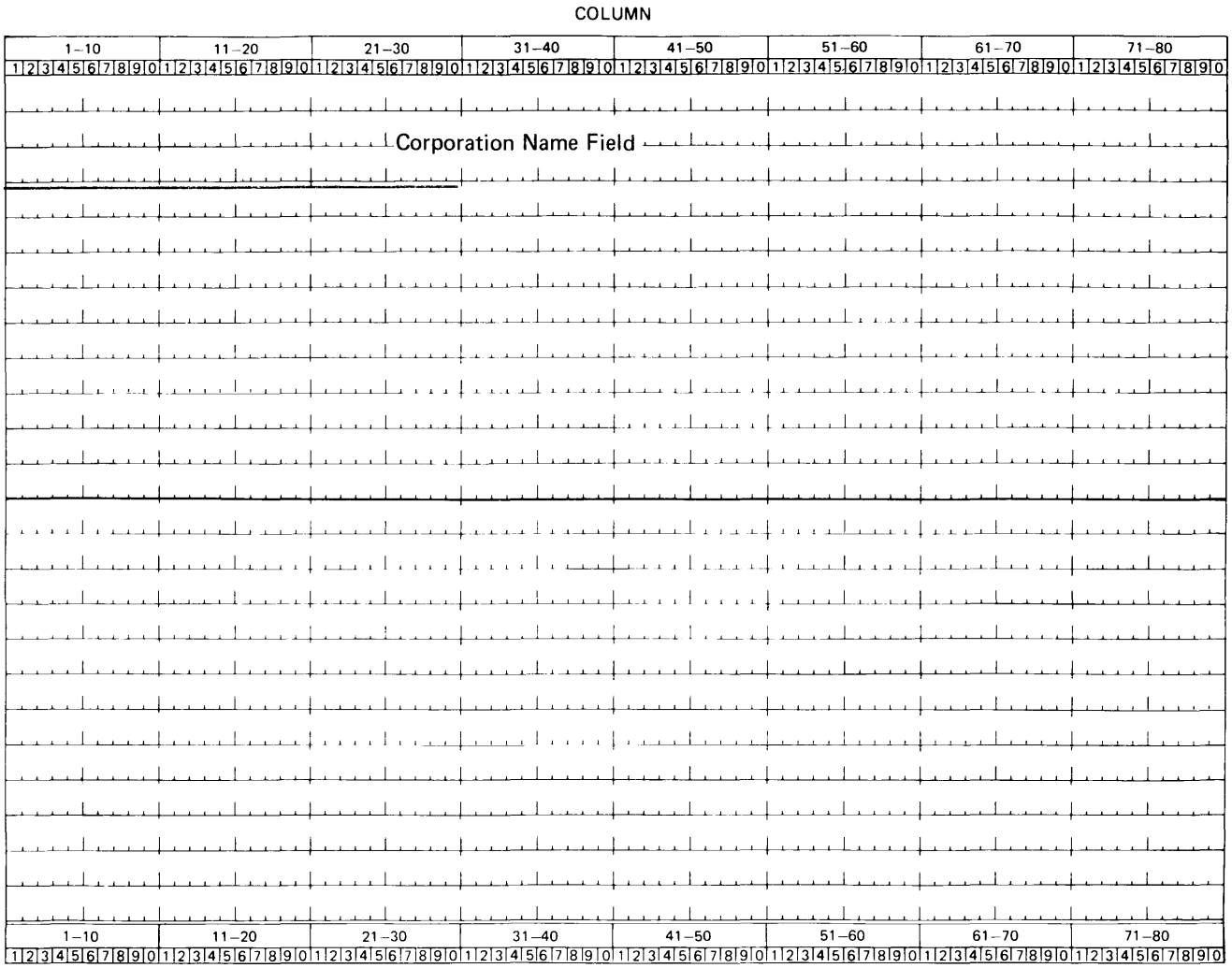
USING THE WORK SHEETS TO DESIGN THE DISPLAYS FOR THE FIRST SAMPLE JOB

The operator's entry will be most efficient if the display designs match the source document. The source documents for this example are the cards in the customer name and address card file.



Begin marking your display contents on your work sheet. The first field should be the corporation name. Look at the work sheet that follows.

Display Screen Layout Sheet



You may place the first field anywhere on the display. The sample places the first field in line 2 and column 1 of the display to match the location of the field on the card. As you begin to mark the location of the field, consider two questions:

- How long should the field be?

  Answer: The instructions from the supervisor show that this field should be 30 positions long.

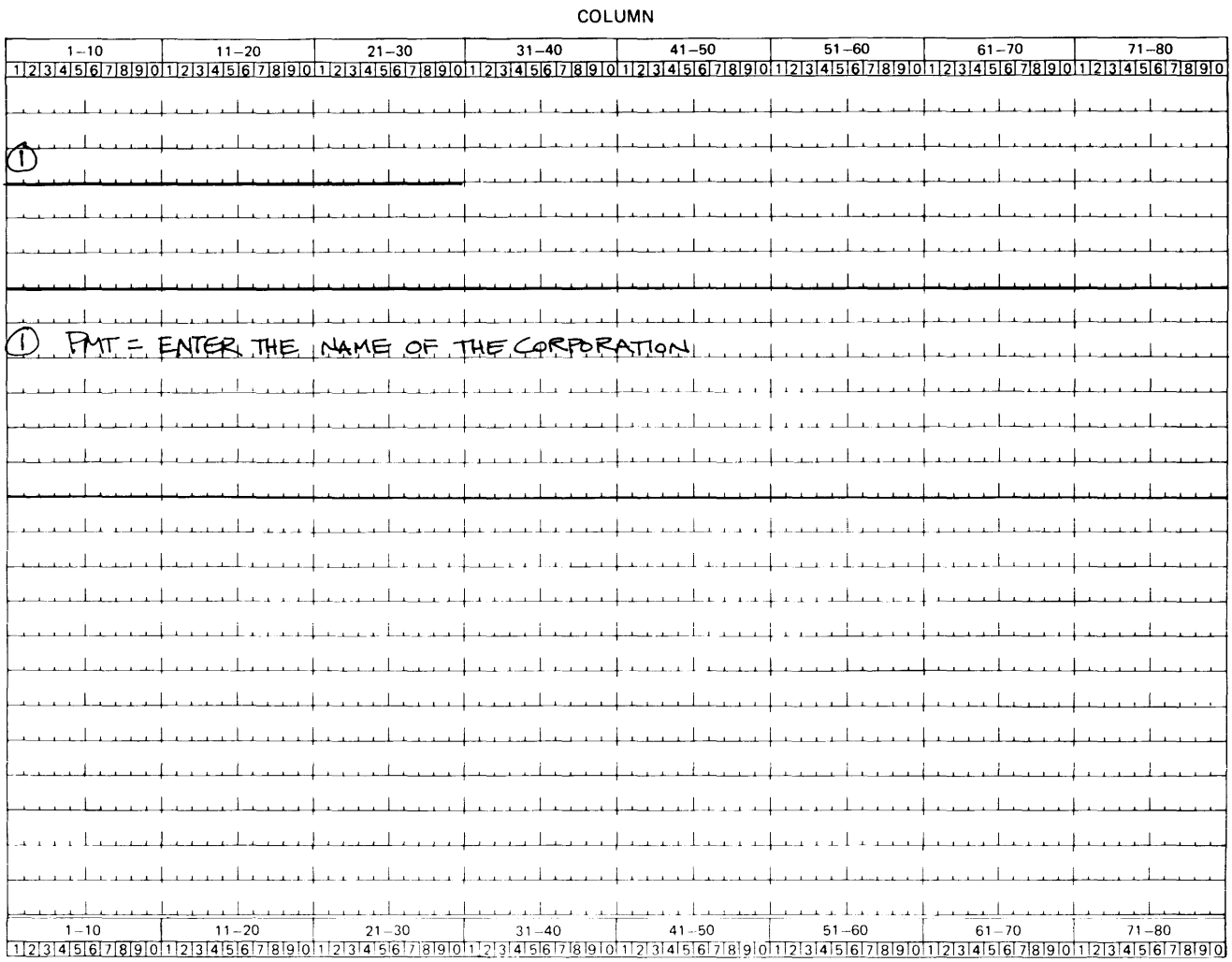
- How will the operator know what to put in the field?

  Answer: Supply information that guides the entry. Two choices are available: the prompt and the literal.

Because the prompt remains on the display only as long as the field is incomplete, it is probably the best choice if other fields will be on the display. Because you know that a number of fields are in the record, you should allow adequate space to include them.

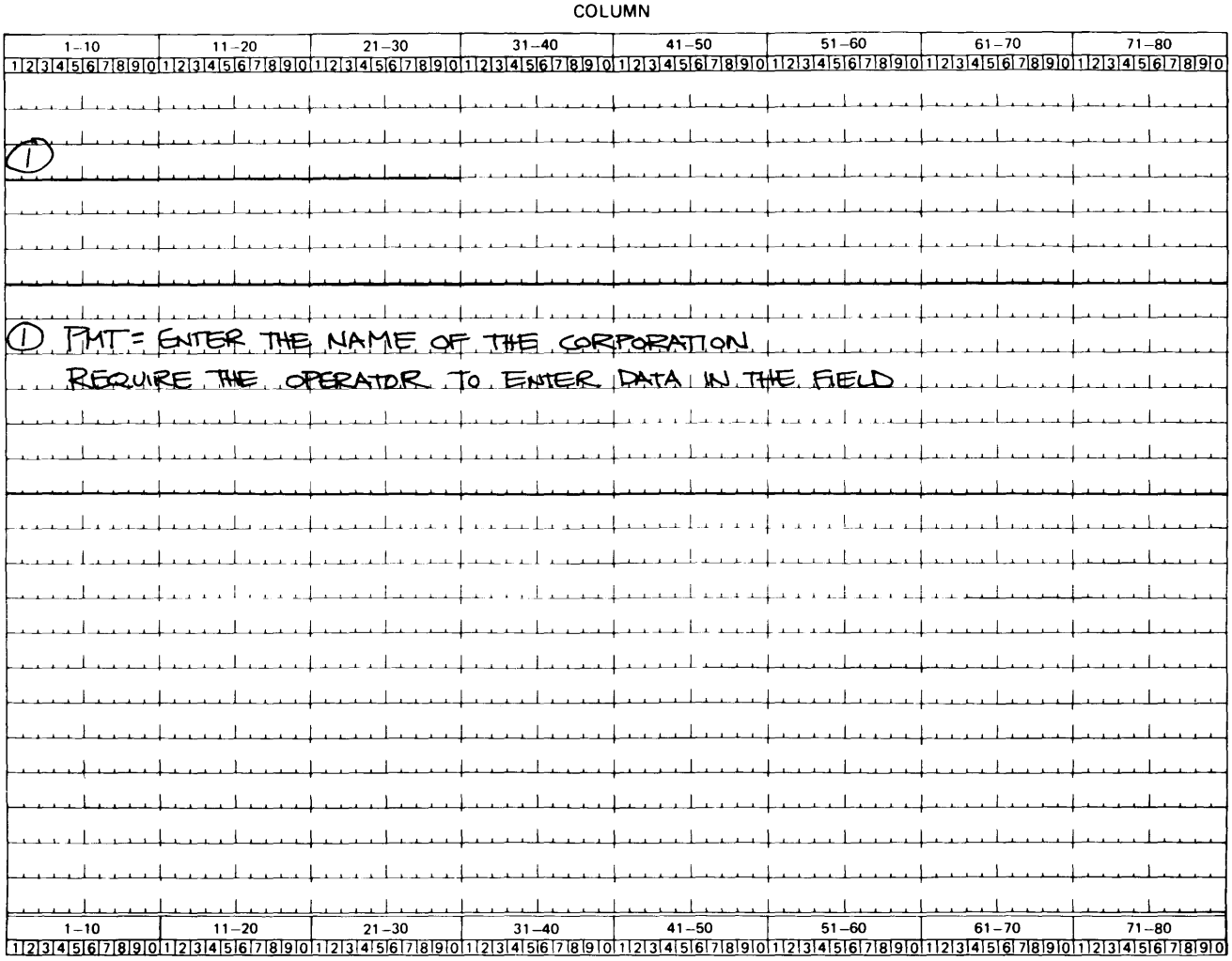
Select the appropriate wording for the prompt and place it on the work sheet. The PMT stands for prompt; the exact wording for the prompt follows. You now have an understandable prompt and a field that the operator can use.

Display Screen Layout Sheet



You should consider one more action before proceeding. The operator should be required to enter data into the first field to guarantee that no record in the customer address data set is without a corporation name field. This is easy to accomplish using DE/RPG. All you need to do at this time is to note on the work sheet that an entry in the field is required. Use the supervisor's instructions (which were provided in the beginning of this chapter) to determine the characteristics of the fields.

Display Screen Layout Sheet



The first field has been defined. You should now have a clear idea of what you want the field to include, where it is located, and the checks and edits you want performed against the entry. You are now ready to define the next field.

The next field contains the customer name. The supervisor's instructions show the length of this field as 30 positions. A prompt is needed for the field. Write this prompt on the work sheet. Consider what restrictions you might want to place on the entry. In addition to requiring that the entry be made as you did for the first field, consider requiring that this entry be alphabetic only. Names generally consist of alphabetic characters and not numeric ones; therefore, specify the field as alphabetic only. Look at the sample work sheet.

Display Screen Layout Sheet

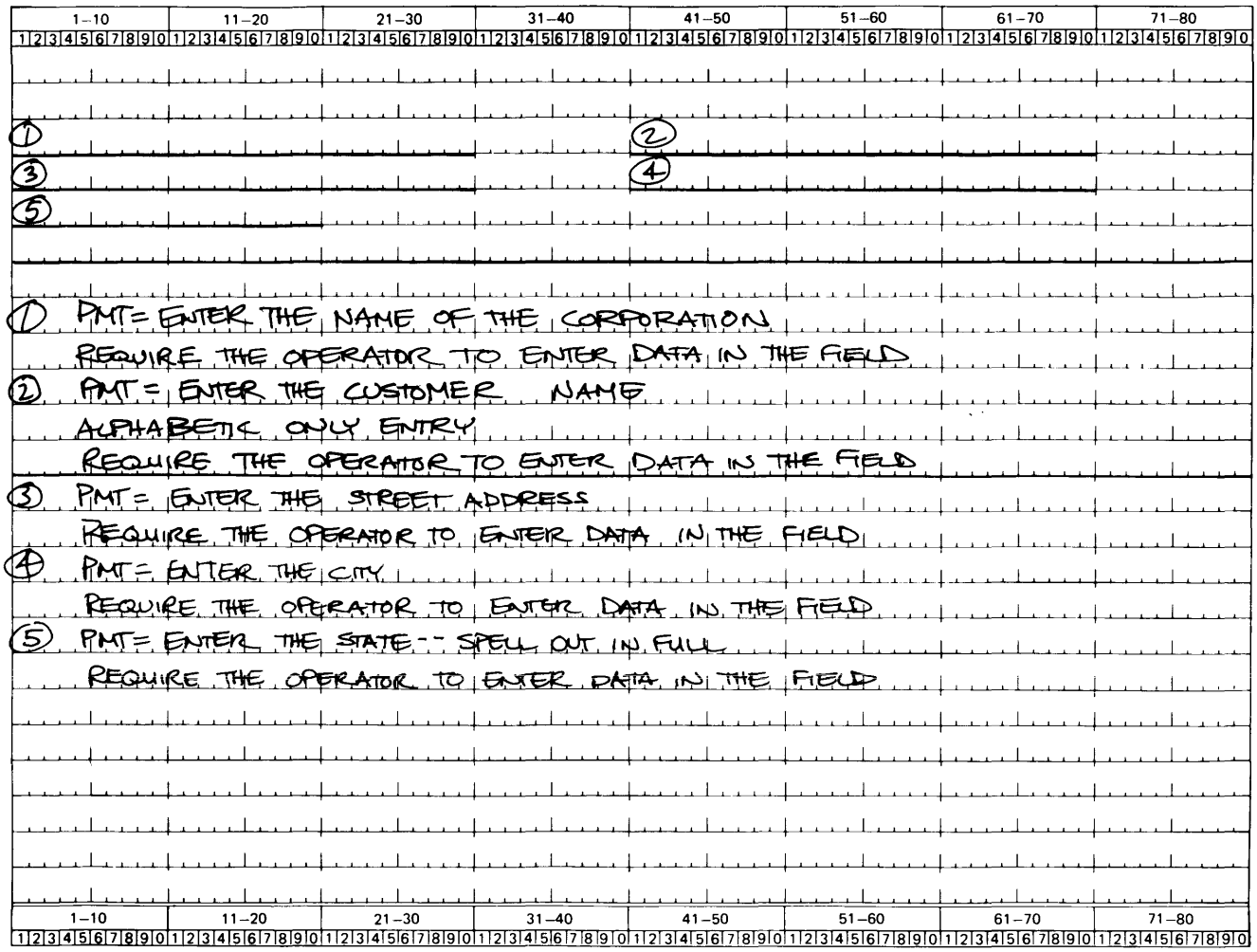
COLUMN

1–10	11–20	21–30	31–40	41–50	51–60	61–70	71–80
1 2 3 4 5 6 7 8 9 0	1 2 3 4 5 6 7 8 9 0	1 2 3 4 5 6 7 8 9 0	1 2 3 4 5 6 7 8 9 0	1 2 3 4 5 6 7 8 9 0	1 2 3 4 5 6 7 8 9 0	1 2 3 4 5 6 7 8 9 0	1 2 3 4 5 6 7 8 9 0
(1)				(2)			
(1) PMT = ENTER THE NAME OF THE CORPORATION							
REQUIRE THE OPERATOR TO ENTER DATA IN THE FIELD							
(2) PMT = ENTER THE CUSTOMER NAME							
ALPHABETIC ONLY ENTRY							
REQUIRE THE OPERATOR TO ENTER DATA IN THE FIELD							
1–10	11–20	21–30	31–40	41–50	51–60	61–70	71 80
1 2 3 4 5 6 7 8 9 0	1 2 3 4 5 6 7 8 9 0	1 2 3 4 5 6 7 8 9 0	1 2 3 4 5 6 7 8 9 0	1 2 3 4 5 6 7 8 9 0	1 2 3 4 5 6 7 8 9 0	1 2 3 4 5 6 7 8 9 0	1 2 3 4 5 6 7 8 9 0

You now have two fields defined; these fields are in the same record. Look again at the source document; you will find that the fields for the street, city, and state are next. Using the same procedure that you used for previous fields, define the street, city, and state fields in the following way.

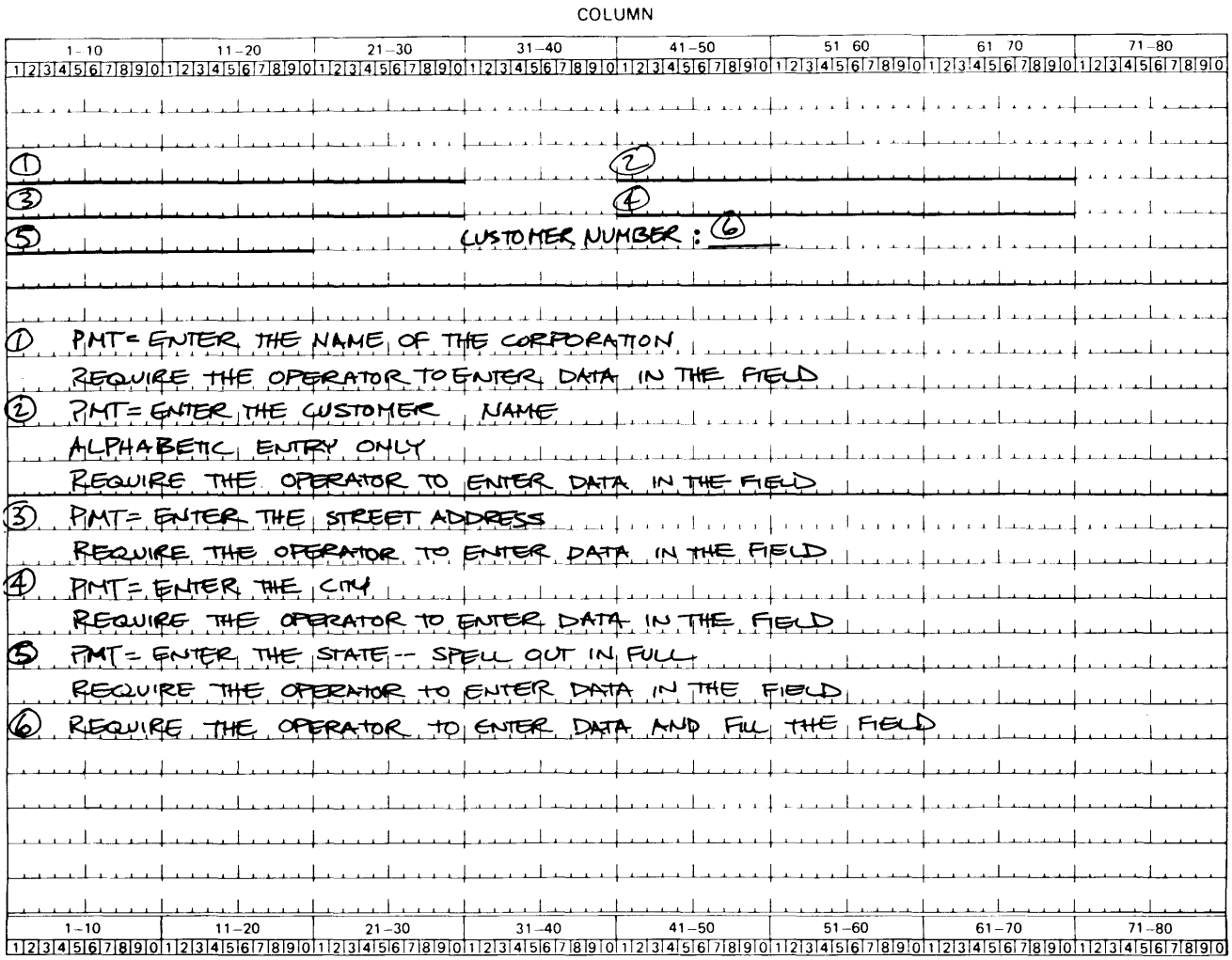
Display Screen Layout Sheet

COLUMN



One of the last fields to define is the customer number field. The operator needs to know what to place in this field. Use the literal *CUSTOMER NUMBER*. If you specify a literal, it will not be written on the diskette, and it will remain on the display to guide the operator until the record advances. There is no particular advantage in using literal in the example. It simply allows you to learn how to use both types of prompting messages.

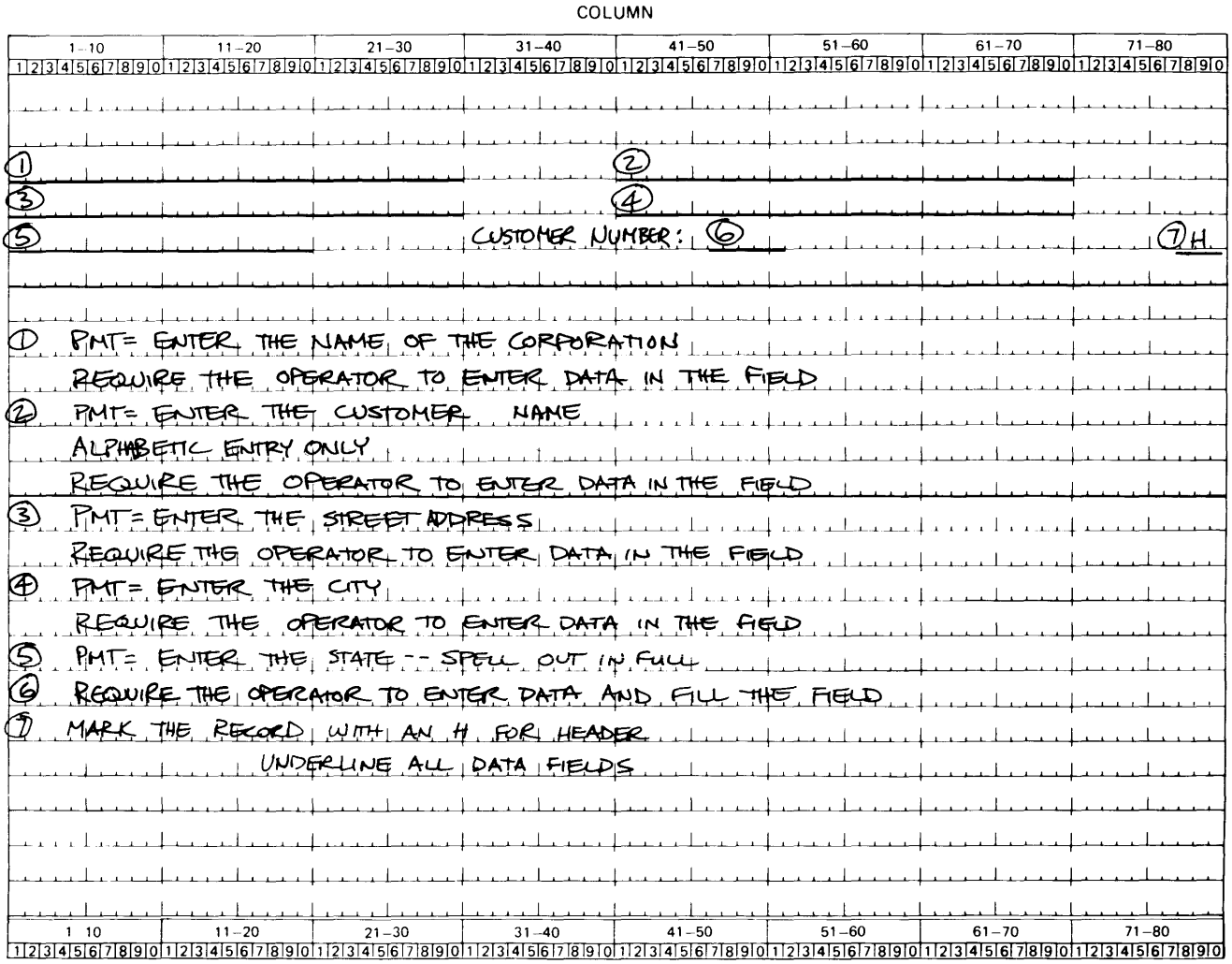
Display Screen Layout Sheet



The literal is separate from the entry field for the customer number. Define the customer number field as you defined previous fields. Consider that all customer numbers are five positions long; require the operator to fill all field positions before exiting from the field.

Next, you must mark the record with an H as instructed by the supervisor. Because this mark must be included in the diskette record, do not use a literal. Remember, literals are not written on the diskette record. Use the insert function. Simply note on the work sheet that the record is to be marked with an H (which will be inserted automatically). The next chapter will describe how to specify an insert.

Display Screen Layout Sheet



All fields have been defined. There is one more thing to consider. The operators will be able to determine how many positions are in each field by looking at a number on the status line. It would be more convenient, however, if you showed them the length of the field on the display. You can do this by using one of the display attributes: column separators (|), underlining (_), or reverse image (■). This example arbitrarily uses the underlining display attribute for all input fields. Be aware that using display attributes requires a display position before and after the field. The position required by the display attributes is not shown on the work sheets.

SUMMARY OF CHAPTER 3

You have designed the displays that the operator will use to enter information from the customer address file.

In this chapter, you have:

- Looked at the source document used for the job
- Received instructions about the formats for the records on the diskette
- Designed display layouts to match the formats of the source document
- Specified checks and edits to be performed against entries in the fields to ensure accurate entry

Try to answer the following questions before you go to the next chapter to learn how to describe the displays on the A specification:

1. Identify the first job. In your own words, describe the instructions using the following fill-in-the-blank entries.

   a. What is the source document that is being used for the job?

   _____

   b. What will the resulting data set consist of?

   _____

2. How many record types are in the job? (1, 2, or 3) _____

3. There are 7 data fields and one literal on the display. Name the data fields and list the literals.

   _____ _____

   _____ _____

   _____ _____

   _____ _____

Check your answers against the answers provided in Appendix A. If you have difficulty answering these questions, reread the chapter.

Chapter 4. Using the A and Z Specifications to Describe the Master Customer Identification Job

The display design for the first example is complete. This design included the placement of the fields on the display and descriptions of the checks and edits or automatic functions to be used for the entries. This chapter describes the entries required on the A and Z specifications to define the display designed in Chapter 3 and to define the job and format characteristics described in the instructions from your supervisor.

You will need blank Z and A specifications and the work sheets you used to define the fields on the display (see Figure 3).

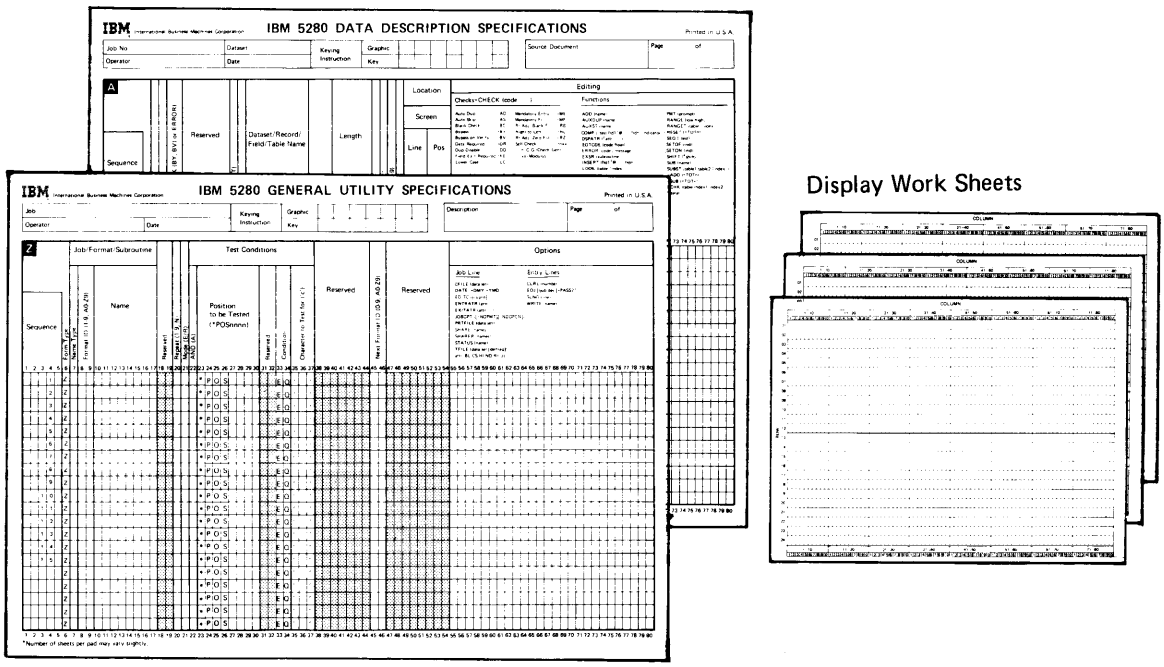


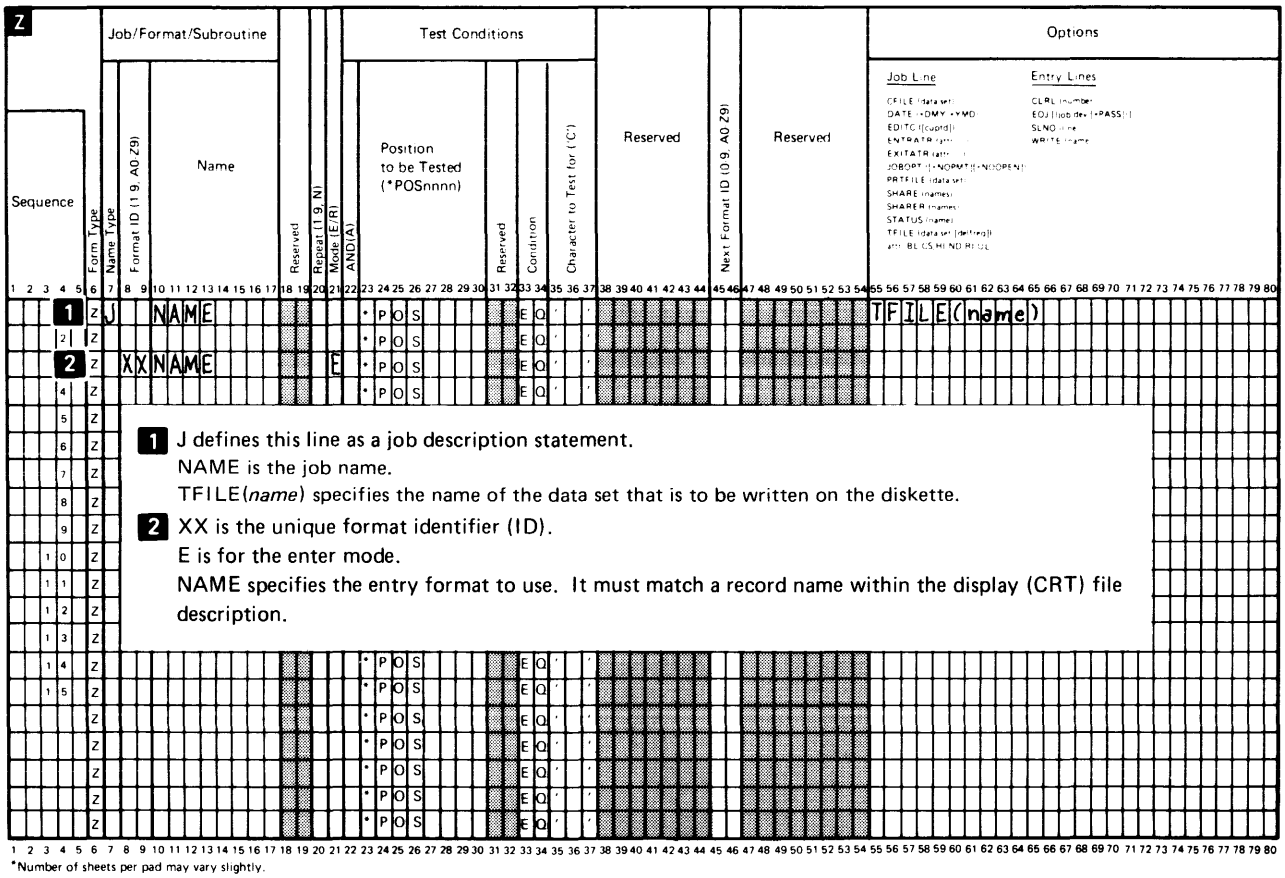
Figure 3. Materials You Will Need to Describe the Job

You will use the Z specification to specify the job and format characteristics. You will use the A specification to describe the display fields, record, and files for this job. A minimum description for each data-entry job is required. Figure 3 illustrates the minimum description.

IBM International Business Machines Corporation

IBM 5280 GENERAL UTILITY SPECIFICATIONS

Job		Keying Instruction	Graphic								Description	Page of
Operator	Date		Key									



*Number of sheets per pad may vary slightly.

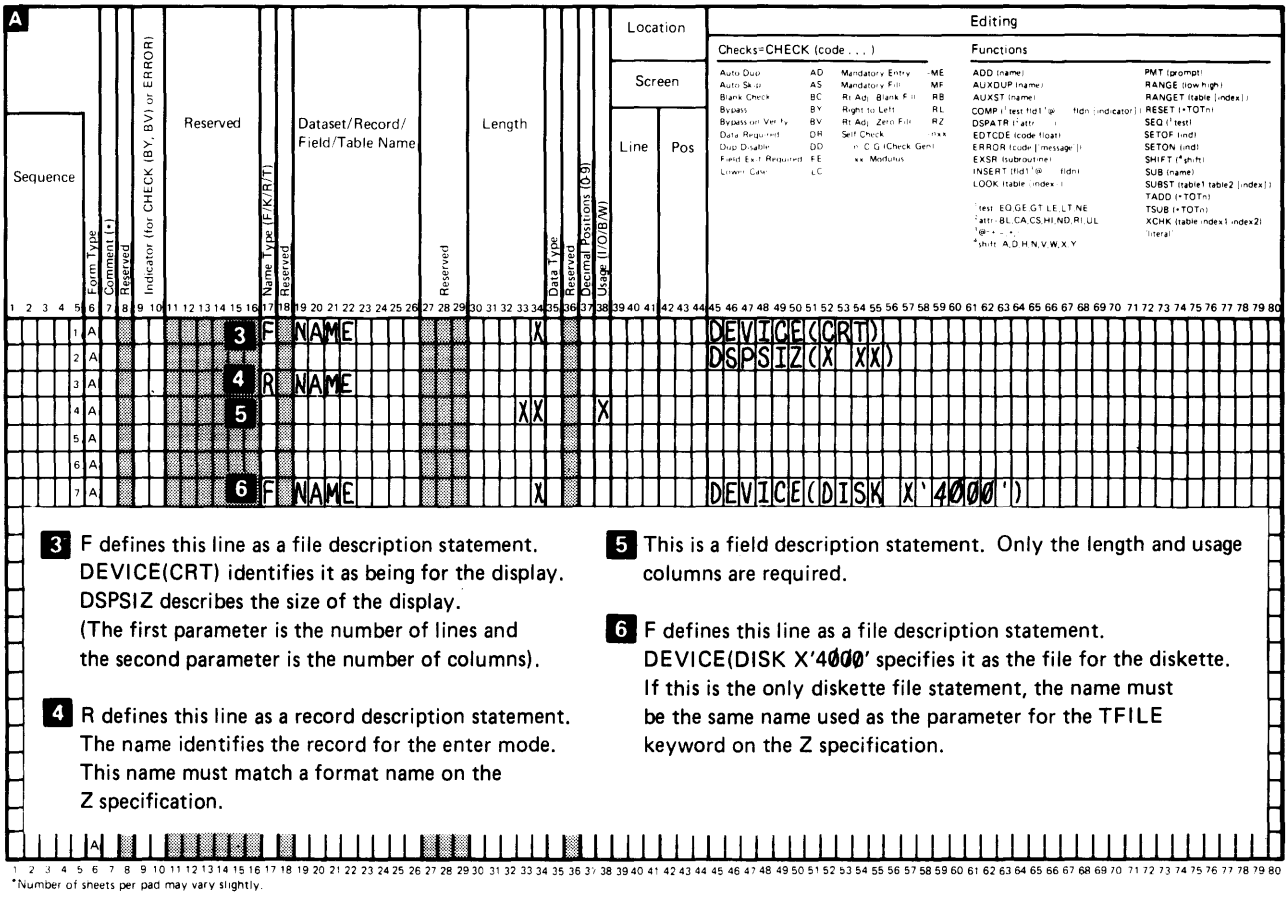
Figure 4 (Part 1 of 2). Sample Program Description

IBM International Business Machines Corporation

IBM 5280 DATA DESCRIPTION SPECIFICATIONS

Printed in U.S.A.

Job No.	Dataset	Keying Instruction	Graphic								Source Document	Page of
Operator	Date		Key									



3 F defines this line as a file description statement. DEVICE(CRT) identifies it as being for the display. DSPSIZ describes the size of the display. (The first parameter is the number of lines and the second parameter is the number of columns).

4 R defines this line as a record description statement. The name identifies the record for the enter mode. This name must match a format name on the Z specification.

5 This is a field description statement. Only the length and usage columns are required.

6 F defines this line as a file description statement. DEVICE(DISK X'4000' specifies it as the file for the diskette. If this is the only diskette file statement, the name must be the same name used as the parameter for the TFILE keyword on the Z specification.

*Number of sheets per pad may vary slightly.

Figure 4 (Part 2 of 2). Sample Program Description

Look at the upper right corner of both the Z and A specifications. This corner contains capitalized words or abbreviations. These *keywords* tell DE/RPG what you want it to do. The information that follows the keyword is called a *parameter*. CHECK(BY) is an editing function in which CHECK is the keyword and BY (which stands for bypass) is the parameter. This editing function fills the associated fields with blanks.

There are a few, simple rules you will have to learn in order to use the keywords. For example, all keywords must be capitalized. They must begin in column 55 of the Z specification and in column 45 of the A specification. The back of the A and Z specification forms lists the keywords and parameters that are available and briefly describes their functions.

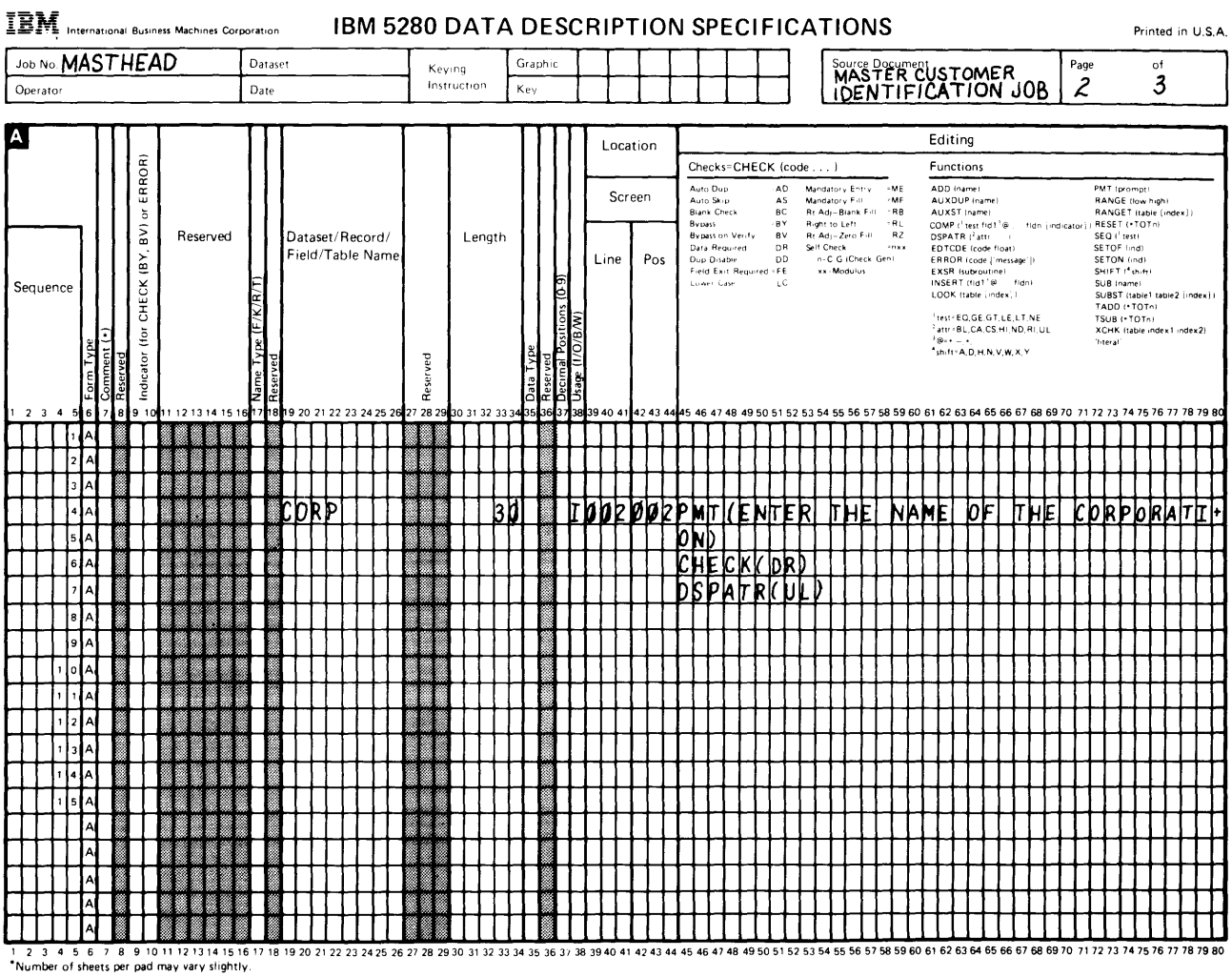
When you have completed this chapter, you should be able to create a job that uses the same format for entry and review and uses the same format to display data and write data on the diskette. The specific concepts you will learn in this chapter are:

- How to specify fields, records, files, formats, and jobs
- How to describe literals, prompts, and inserts
- How to describe checks and edits
- How to mark a record type for later identification

DESCRIBING THE FIELDS ON THE A SPECIFICATION

For this example, begin on the fourth line of your A specification. This will be the location of your first field description. A file and record description must precede the field description; by beginning on line 4, you are leaving space to include these later. You are starting with the field descriptions rather than the record or file description because the display fields have already been designed on the work sheets. Once the fields are described on the A specification, it is simple to add the necessary record and file descriptions.

As you look at the work sheet for the display, notice that the first field is the corporation name. Review the supervisor's instructions and your entries on the work sheet; the field name is CORP. Starting at column 19 on the A specification, write the field name. It is not necessary to name fields if they will not be referenced elsewhere. Naming fields increases the program size. Field names can be no more than 6 characters long. Refer to the following sample as you learn what entries are needed to specify this field.



IBM International Business Machines Corporation — IBM 5280 DATA DESCRIPTION SPECIFICATIONS — Printed in U.S.A.

Job No. MASTHEAD | Dataset | Keying Instruction | Graphic | Key
Operator | Date

Source Document: MASTER CUSTOMER IDENTIFICATION JOB | Page 2 of 3

A

Sequence	Form Type	Comment (*)	Reserved	Indicator (for CHECK (BY, BV) or ERROR)	Reserved	Name Type (F/K/R/T)	Reserved	Dataset/Record/ Field/Table Name	Reserved	Length	Data Type	Reserved	Decimal Positions (0-9)	Usage (I/O/B/W)	Location: Screen Line	Location: Screen Pos	Editing
4	A							CORP		30				I	002	002	PMT(ENTER THE NAME OF THE CORPORATI+
																	ON)
																	CHECK(DR)
																	DSPATR(UL)

Editing — Checks=CHECK (code . . .)

Auto Dup	AD	Mandatory Entry	*ME
Auto Skip	AS	Mandatory Fill	*MF
Blank Check	BC	Rt Adj–Blank Fill	*RB
Bypass	BY	Right to Left	*RL
Bypass on Verify	BV	Rt Adj–Zero Fill	*RZ
Data Required	DR	Self Check	*mxx
Dup Disable	DD	n-C G (Check Gen)	
Field Exit Required	*FE	xx-Modulus	
Lower Case	LC		

Editing — Functions

ADD (name)	PMT (prompt)
AUXDUP (name)	RANGE (low high)
AUXST (name)	RANGET (table [index])
COMP (test fld1'@ fldn [indicator])	RESET (+TOTn)
DSPATR (1attr ,)	SEQ (test)
EDTCDE (code [float])	SETOF (ind)
ERROR (code [,message])	SETON (ind)
EXSR (subroutine)	SHIFT (*shift)
INSERT (fld1'@ fldn)	SUB (name)
LOOK (table ,index,)	SUBST (table1 table2 [index])
	TADD (+TOTn)
1test=EQ,GE,GT,LE,LT,NE	TSUB (+TOTn)
2attr=BL,CA,CS,HI,ND,RI,UL	XCHK (table index1 index2)
3@=+ – *	'literal'
*shift=A,D,H,N,V,W,X,Y	

*Number of sheets per pad may vary slightly.

Next, look at columns 30 through 34. These columns specify the field length. This number (30) must be right-adjusted (which means that it must end in column 34). Because you did not specify that the entry be a certain data type (such as numeric only) or that it be used in calculations, leave columns 35 through 37 blank. A blank is equivalent to an alphabetic field.

Column 38 is the usage column. Each field definition must have an entry in this column. There are three choices for basic data entry: I (input), O (output), and W (work space). *I* specifies that the field will be displayed during enter and review modes. It also allows the operator to enter data into the field and allows the field to be written in a diskette record. *O* specifies that the field will be displayed, but cannot be altered by the operator or written in a diskette record. *W* prevents the field from being displayed, altered by the operator, or written in the diskette record; it can be specified, for example, for fields that contain intermediate calculation results. Because this sample program requires that the operator enter data into this field and have that data written in the diskette record, the entry in column 38 must be I.

Columns (39 through 44) specify the position of the field on the display. Although leading 0's are used in this example, they are not required by DE/RPG. The numbers entered in columns 39 through 44 determine the line and column positions on the display. If you do not use these columns, the first field is in line 3 column 1 and all fields that follow are strung together in succession across the display. The location of a field on the display is up to you. Although this field began in column 1 on the display work sheet, it must be positioned in column 2 in the program. Because the field met the following conditions, its display attribute would have been ignored once the field was exited: The field was (1) on line 2 column 1, (2) used with a prompt, and (3) used a display attribute. To preserve the underline attribute and still be able to use a prompt, the field must be positioned on line 2, column 2.

Columns 45 through 80 contain the prompts, literals, automatic functions, and checks and edits for the field. The keyword PMT (which stands for prompt) must be used for all messages you want displayed on line 1 (row 2) of the display. The wording for the prompt follows the keyword and must be enclosed in parentheses.

Notice that the prompt wording exceeds the space provided by columns 45 through 80. The + continuation character allows you to continue the wording to the next line on the A specification. When you continue a field description, columns 7 through 44 of the subsequent lines must be blank. The two continuation characters are + and -. Basically, the difference is that DE/RPG does not count beginning blanks starting in position 45 on the next line when the + continuation character is used. See the *DE/RPG Reference Manual* for additional details about continuation characters.

The first edit, CHECK(DR), is on the line following the wording for the prompt. This line skipping is not required by DE/RPG. Only a blank between keywords and parameters is necessary. Line skipping is used in the example, because it makes the program easier to read. No continuation character is required because the keyword and parameter are complete. The CHECK(DR) keyword and parameter satisfy the requirements for a field that must have operator entry. DR means data required; the operator must enter at least one nonblank character into the field. The DSPATR(UL) keyword and parameter stand for display attribute underline. This keyword and parameter will cause the system to underline all positions of the field to identify the field length to the operator. The first field description is complete.

The next field is the customer name. The name of the field is CUSN.

IBM International Business Machines Corporation

IBM 5280 DATA DESCRIPTION SPECIFICATIONS

Printed in U.S.A.

Job No. MASTHEAD	Dataset	Keying Instruction	Graphic								Source Document MASTER CUSTOMER IDENTIFICATION JOB	Page 2	of 3
Operator	Date		Key										

A

Sequence	Form Type	Comment (*)	Reserved	Indicator (for CHECK (BY, BV) or ERROR)	Reserved	Name Type (F/K/R/T)	Reserved	Dataset/Record/ Field/Table Name	Reserved	Length	Data Type	Reserved	Decimal Positions (0-9)	Usage (I/O/B/W)	Location: Screen Line	Location: Screen Pos	Editing
1 2 3 4 5	6	7	8	9 10	11 12 13 14 15 16	17	18	19 20 21 22 23 24 25 26	27 28 29	30 31 32 33 34	35	36	37	38	39 40 41	42 43 44	45 46 47 48 49 50 51 52 53 54 55 56 57 58 59 60 61 62 63 64 65 66 67 68 69 70 71 72 73 74 75 76 77 78 79 80
1	A																
2	A																
3	A																
4	A							CORP		30				I	002	002	PMT(ENTER THE NAME OF THE CORPORATI+
5	A																ON)
6	A																CHECK(DR)
7	A																DSPATR(UL)
8	A							CUSN		30X				I	002	041	PMT(ENTER THE CUSTOMER NAME)
9	A																CHECK(DR)
10	A																DSPATR(UL)
11	A																
12	A																
13	A																
14	A																
15	A																
	A																
	A																
	A																
	A																
	A																

Checks=CHECK (code . . .)

Auto Dup	-AD	Mandatory Entry	=ME
Auto Skip	-AS	Mandatory Fill	=MF
Blank Check	-BC	Rt Adj–Blank Fill	=RB
Bypass	=BY	Right to Left	=RL
Bypass on Verify	-BV	Rt Adj–Zero Fill	=RZ
Data Required	-DR	Self Check	=nxx
Dup Disable	-DD	n-C,G (Check Gen)	
Field Exit Required	-FE	xx=Modulus	
Lower Case	-LC		

Functions

ADD (name)	PMT (prompt)
AUXDUP (name)	RANGE (low high)
AUXST (name)	RANGET (table {index})
COMP ([1]test fld1[2]@ fldn {indicator})	RESET (+TOTn)
DSPATR ([2]attr [1])	SEQ ([1]test)
EDTCDE (code {out})	SETOF (ind)
ERROR (code {'message'})	SETON (ind)
EXSR (subroutine)	SHIFT ([4]shift)
INSERT (fld1[2]@ fldn)	SUB (name)
LOOK (table {index})	SUBST (table1 table2 {index})
[1]test=EQ,GE,GT,LE,LT,NE	TADD (+TOTn)
[2]attr=BL,CA,CS,HI,ND,RI,UL	TSUB (+TOTn)
[3]@=+,–,*,/	XCHK (table index1 index2)
[4]shift=A,D,H,N,V,W,X,Y	'literal'

1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20 21 22 23 24 25 26 27 28 29 30 31 32 33 34 35 36 37 38 39 40 41 42 43 44 45 46 47 48 49 50 51 52 53 54 55 56 57 58 59 60 61 62 63 64 65 66 67 68 69 70 71 72 73 74 75 76 77 78 79 80

*Number of sheets per pad may vary slightly.

Notice that all descriptions, except the entry in column 35, are similar to those for the first field. Column 35 defines the data type that is acceptable for field entry. Remember that you noted that this field should be alphabetic only. The X in this column specifies that all entries be edited to accept only alphabetic characters. The *DE/RPG Reference Manual* contains descriptions of the possible entries in this column.

The next three fields (STREET, CITY, and STATE) are similar to the preceding fields. Try to describe them on the A specification without looking at the sample.

IBM International Business Machines Corporation

IBM 5280 DATA DESCRIPTION SPECIFICATIONS

Printed in U.S.A.

Job No. MASTHEAD	Dataset	Keying Instruction	Graphic								Source Document: MASTER CUSTOMER IDENTIFICATION JOB	Page 2	of 3
Operator	Date		Key										

A

Sequence	Form Type	Comment (*)	Reserved	Indicator (for CHECK (BY, BV) or ERROR)	Reserved	Name Type (F/K/R/T)	Reserved	Dataset/Record/Field/Table Name	Reserved	Length	Data Type	Reserved	Decimal Positions (0-9)	Usage (I/O/B/W)	Location: Screen Line	Location: Screen Pos	Editing
1 2 3 4 5	6	7	8	9 10	11 12 13 14 15 16	17	18	19 20 21 22 23 24 25 26	27 28 29	30 31 32 33 34	35	36	37	38	39 40 41	42 43 44	45 46 47 48 49 50 51 52 53 54 55 56 57 58 59 60 61 62 63 64 65 66 67 68 69 70 71 72 73 74 75 76 77 78 79 80
1	A																
2	A																
3	A																
4	A							CORP		30				I	002	002	PMT(ENTER THE NAME OF THE CORPORATI+
5	A																ON)
6	A																CHECK(DR)
7	A																DSPATR(UL)
8	A							CUSN		30	X			I	002	041	PMT(ENTER THE CUSTOMER NAME)
9	A																CHECK(DR)
10	A																DSPATR(UL)
11	A							STREET		30				I	003	001	PMT(ENTER THE STREET ADDRESS)
12	A																CHECK(DR)
13	A																DSPATR(UL)
14	A							CITY		30				I	003	041	PMT(ENTER THE CITY)
15	A																CHECK(DR)
	A																DSPATR(UL)
	A							STATE		20				I	004	001	PMT(ENTER THE STATE--SPELL OUT IN F+
	A																ULL)
	A																CHECK(DR)
	A																DSPATR(UL)

1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20 21 22 23 24 25 26 27 28 29 30 31 32 33 34 35 36 37 38 39 40 41 42 43 44 45 46 47 48 49 50 51 52 53 54 55 56 57 58 59 60 61 62 63 64 65 66 67 68 69 70 71 72 73 74 75 76 77 78 79 80

*Number of sheets per pad may vary slightly.

The next field does not have a name. It is the literal for the customer number field. Notice that this field has no length specified and that it has an O in column 38. Literals do not have length entries. They are always O (output) usage because they are not written in the diskette record. The wording for the literal is within apostrophes (') which tell DE/RPG that this is a message. All literal messages must be enclosed in apostrophes.

IBM International Business Machines Corporation

IBM 5280 DATA DESCRIPTION SPECIFICATIONS

Printed in U.S.A.

Job No. MASTHEAD	Dataset	Keying Instruction	Graphic
Operator	Date		Key

Source Document: MASTER CUSTOMER IDENTIFICATION JOB | Page 3 of 3

A

Sequence	Form Type	Comment (*)	Reserved	Indicator (for CHECK (BY, BV) or ERROR)	Reserved	Name Type (F/K/R/T)	Reserved	Dataset/Record/ Field/Table Name	Reserved	Length	Data Type	Reserved	Decimal Positions (0-9)	Usage (I/O/B/W)	Location: Line	Location: Pos	Editing
1	A													O	04	03	0'CUSTOMER NUMBER:'
2	A																
3	A																
4	A																
5	A																
6	A																
7	A																
8	A																
9	A																
10	A																
11	A																
12	A																
13	A																
14	A																
15	A																
	A																
	A																
	A																
	A																
	A																

Checks=CHECK (code . . .)

Auto Dup	AD	Mandatory Entry	ME
Auto Skip	AS	Mandatory Fill	MF
Blank Check	BC	Rt Adj-Blank Fill	RB
Bypass	BY	Right to Left	RL
Bypass on Verify	BV	Rt Adj-Zero Fill	RZ
Data Required	DR	Self Check	nxx
Dup Disable	DD	n-C,G (Check Gen)	
Field Exit Required	FE	xx-Modulus	
Lower Case	LC		

Functions

ADD (name)	PMT (prompt)
AUXDUP (name)	RANGE (low high)
AUXST (name)	RANGET (table [index:])
COMP (1test fld1 1@ fldn [indicator])	RESET (+TOTn)
DSPATR (2attr . .)	SEQ (1test)
EDTCDE (code [float])	SETOF (ind)
ERROR (code ['message'])	SETON (ind)
EXSR (subroutine)	SHIFT (4shift)
INSERT (fld1 1@ fldn)	SUB (name)
LOOK (table [index:])	SUBST (table1 table2 [index:])
	TADD (+TOTn)
1test-EQ,GE,GT,LE,LT,NE	TSUB (+TOTn)
2attr-BL,CA,CS,HI,ND,RI,UL	XCHK (table index1 [index2])
3@-+ - .	'literal'
4shift-A,D,H,N,V,W,X,Y	

1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20 21 22 23 24 25 26 27 28 29 30 31 32 33 34 35 36 37 38 39 40 41 42 43 44 45 46 47 48 49 50 51 52 53 54 55 56 57 58 59 60 61 62 63 64 65 66 67 68 69 70 71 72 73 74 75 76 77 78 79 80

*Number of sheets per pad may vary slightly.

The customer number entry field is next. It introduces one new function: BC. BC stands for blank check. Along with DR, this parameter requires the operator to enter data with no blanks in the entire field. Because both the DR and BC parameters are related to the CHECK keyword, they can be grouped within the same parentheses. However, they must have a blank between them. These parameters require the operator to both enter and fill the field.

IBM International Business Machines Corporation

IBM 5280 DATA DESCRIPTION SPECIFICATIONS

Printed in U.S.A.

Job No MASTHEAD	Dataset	Keying Instruction	Graphic								Source Document MASTER CUSTOMER IDENTIFICATION JOB	Page 3	of 3
Operator	Date		Key										

A

Sequence	Form Type	Comment (*)	Reserved	Indicator (for CHECK (BY, BV) or ERROR)	Reserved	Name Type (F/K/R/T)	Reserved	Dataset/Record/ Field/Table Name	Reserved	Length	Data Type	Reserved	Decimal Positions (0-9)	Usage (I/O/B/W)	Location: Screen Line	Location: Screen Pos	Editing: Checks-CHECK (code . . .) / Functions
1	A														0004	030	'.CUSTOMER NUMBER:'
2	A							NUMBER		5				I	0004	047	CHECK(DR BC)
3	A																DSPATR(UL)
4	A																
5	A																
6	A																
7	A																
8	A																
9	A																
10	A																
11	A																
12	A																
13	A																
14	A																
15	A																
	A																
	A																
	A																
	A																
	A																

Checks-CHECK (code . . .): Auto Dup AD; Auto Skip AS; Blank Check BC; Bypass BY; Bypass on Verify BV; Data Required DR; Dup Disable DD; Field Exit Required FE; Lower Case LC; Mandatory Entry ME; Mandatory Fill MF; Rt Adj-Blank Fill RB; Right to Left RL; Rt Adj-Zero Fill RZ; Self Check nn (n-C,G (Check Gen) xx-Modulus

Functions: ADD (name); AUXDUP (name); AUXST (name); COMP (1test fld1 2@ fldn [indicator]); DSPATR (1attr); EDTCDE (code [float]); ERROR (code [message]); EXSR (subroutine); INSERT (fld1 2@ fldn); LOOK (table [index]); PMT (prompt); RANGE (low high); RANGET (table [index]); RESET (+TOTn); SEQ (1test); SETOF (ind); SETON (ind); SHIFT (4shift); SUB (name); SUBST (table1 table2 [index]); TADD (+TOTn); TSUB (+TOTn); XCHK (table index1 index2); literal

1test=EQ,GE,GT,LE,LT,NE; 2attr=BL,CA,CS,HI,ND,RI,UL; 3@=+,-,x,/; 4shift=A,D,H,N,V,W,X,Y

*Number of sheets per pad may vary slightly.

The next field to define after the customer number field is a field that contains no data. This field was not defined on the work sheet. You know the field is required because the supervisor's instructions told you to place the H marker in the last position of the diskette record. These instructions also told you that the record is 150 positions long. If you add the fields that have been defined for the record, you will find the total is 146. Because the H is to be in position 150, a field of blanks is needed to pad the records.

The CHECK(BY) keyword and parameter will leave a field of blanks with a specified length in the diskette record without operator entry. BY stands for bypass. The result is that the H in the next field is automatically inserted into position 150 of the diskette record.

To determine that the H will be in position 150 of the record, add the entries in the length columns for each field. Next, subtract the total from the mark position 150; this calculation will tell you the length of the blank field.

```
 30 Corporation name
 30 Customer name
 30 Street
 30 City
 20 State
+ 5 Customer number
-----
145 Total field positions

149 Positions available up
    to last postition
- 145 Total field positions
-----
  4 Blanks needed
```

Therefore, the length entry in the column for the CHECK(BY) field should be 4.

IBM International Business Machines Corporation

IBM 5280 DATA DESCRIPTION SPECIFICATIONS

Printed in U.S.A.

Job No MASTHEAD	Dataset	Keying Instruction	Graphic								
Operator	Date		Key								

Source Document	Page	of
MASTER CUSTOMER IDENTIFICATION JOB	3	3

A

Sequence	Form Type	Comment (*)	Reserved	Indicator (for CHECK (BY, BV) or ERROR)	Reserved	Name Type (F/K/R/T)	Reserved	Dataset/Record/ Field/Table Name	Reserved	Length	Data Type	Reserved	Decimal Positions (0-9)	Usage (I/O/B/M)	Location: Screen Line	Location: Screen Pos	Editing: Checks=CHECK (code . . .)	Editing: Functions
1	A																	
2	A							NUMBER		5				I	00	40	47 CHECK(DR BC)	
3	A																DSPATR(UL)	
4	A									4				I			CHECK(BY)	
5	A																	
6	A																	
7	A																	
8	A																	
9	A																	
10	A																	
11	A																	
12	A																	
13	A																	
14	A																	
15	A																	
	A																	
	A																	
	A																	
	A																	
	A																	

1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20 21 22 23 24 25 26 27 28 29 30 31 32 33 34 35 36 37 38 39 40 41 42 43 44 45 46 47 48 49 50 51 52 53 54 55 56 57 58 59 60 61 62 63 64 65 66 67 68 69 70 71 72 73 74 75 76 77 78 79 80

*Number of sheets per pad may vary slightly.

The last field is the marker (H) for the record type. Use the insert function to place the H in the diskette record.

IBM International Business Machines Corporation — IBM 5280 DATA DESCRIPTION SPECIFICATIONS — Printed in U.S.A.

Job No MASTHEAD	Dataset	Keying Instruction	Graphic								Source Document MASTER CUSTOMER IDENTIFICATION JOB	Page 3	of 3
Operator	Date		Key										

Sequence	Form Type	Comment (*)	Reserved	Indicator (for CHECK (BY, BV) or ERROR)	Reserved	Name Type (F/K/R/T)	Reserved	Dataset/Record/ Field/Table Name	Reserved	Length	Data Type	Reserved	Decimal Positions (0-9)	Usage (I/O/B/W)	Location: Screen Line	Location: Screen Pos	Editing: Checks=CHECK (code . . .)	Editing: Functions
1	A														00	40	'00 4030 'CUSTOMER NUMBER:'	
2	A							NUMBER		5				I	00	40	47CHECK(DR BC)	
3	A																DSPATR(UL)	
4	A									4				I			CHECK(BY)	
5	A							MARK		1				I	00	40	80INSERT('H')	
6	A																	
7	A																	
8	A																	
9	A																	
10	A																	
11	A																	
12	A																	
13	A																	
14	A																	
15	A																	
	A																	
	A																	
	A																	
	A																	
	A																	

Checks=CHECK (code . . .): Auto Dup -AD; Auto Skip -AS; Blank Check -BC; Bypass -BY; Bypass on Verify -BV; Data Required -DR; Dup Disallow -DD; Field Exit Required -FE; Lower Case -LC; Mandatory Entry -ME; Mandatory Fill -MF; Rt Adj-Blank Fill -RB; Right to Left -RL; Rt Adj-Zero Fill -RZ; Self Check -nxx; n-C,G (Check Gen); xx Modulus

Functions: ADD (name); AUXDUP (name); AUXST (name); COMP (1 test fld1 1@ fldn indicator); DSPATR (2 attr); EDTCDE (code [float]); ERROR (code [/'message']); EXSR (subroutine); INSERT (fld1 1@ fldn); LOOK (table [index] [ ]); PMT (prompt); RANGE (low high); RANGET (table [index] [ ]); RESET (+TOTn); SEQ (1 test); SETOF (ind); SETON (ind); SHIFT (4 shift); SUB (name); SUBST (table1 table2 [index] [ ]); TADD (+TOTn); TSUB (+TOTn); XCHK (table index1 index2); 'literal'

1 test=EQ,GE,GT,LE,LT,NE; 2 attr=BL,CA,CS,HI,ND,RI,UL; 3 @-+,*,/; 4 shift=A,D,H,N,V,W,X,Y

*Number of sheets per pad may vary slightly.

The usage column contains an I because you want this mark written in the *diskette* record. The insert function automatically includes the character H in the record without operator entry. Notice that the H is enclosed in apostrophes. The apostrophes mean that the H is character data. Only named fields, arithmetic expressions, or constants (either all character or all numeric) can be used with the insert function. The *DE/RPG Reference Manual* describes the insert function in detail.

You have completed the description of the fields and have defined all the checks, edits, and automatic functions required for the job. The next step is to describe the record.

DESCRIBING THE RECORD ON THE A SPECIFICATION

The third line of the A specification is the location of the record description. Place an R in column 17. The R identifies the following field descriptions as a description of one record. No entry appears in this column for the field descriptions. The only other information that is required for the record description is a name. This name must begin in column 19 and can be no longer than 8 characters. The example uses the name HEADER to identify the record.

IBM International Business Machines Corporation — IBM 5280 DATA DESCRIPTION SPECIFICATIONS — Printed in U.S.A.

Job No. MASTHEAD	Dataset	Keying Instruction	Graphic	Source Document MASTER CUSTOMER IDENTIFICATION JOB	Page 2 of 3
Operator	Date		Key		

Sequence	Form Type	Comment (*)	Reserved	Indicator (for CHECK (BY, BV) or ERROR)	Reserved	Name Type (F/K/R/T)	Reserved	Dataset/Record/Field/Table Name	Reserved	Length	Data Type	Reserved	Decimal Positions (0-9)	Usage (I/O/B/W)	Location: Screen Line	Location: Screen Pos	Editing: Checks=CHECK (code . . .) / Functions
1 2 3 4 5	6	7	8	9 10	11 12 13 14 15 16	17	18	19 20 21 22 23 24 25 26	27 28 29	30 31 32 33 34	35	36	37	38	39 40 41	42 43 44	45 46 47 48 49 50 51 52 53 54 55 56 57 58 59 60 61 62 63 64 65 66 67 68 69 70 71 72 73 74 75 76 77 78 79 80
1	A																
2	A																
3	A					R		HEADER									
4	A							CORP		30				I	002	002	PMT(ENTER THE NAME OF THE CORPORATI+
5	A																ON)
6	A																CHECK(DR)
7	A																DSPATR(UL)
8	A							CUSN		30	X			I	002	041	PMT(ENTER THE CUSTOMER NAME)
9	A																CHECK(DR)
10	A																DSPATR(UL)
11	A							STREET		30				I	003	001	PMT(ENTER THE STREET ADDRESS
12	A																CHECK(DR)
13	A																DSPATR(UL)
14	A							CITY		30				I	003	041	PMT(ENTER THE CITY)
15	A																CHECK(DR)
	A																DSPATR(UL)
	A							STATE		20				I	004	001	PMT(ENTER THE STATE--SPELL OUT IN F+
	A																ULL)
	A																CHECK(DR)
	A																DSPATR(UL)

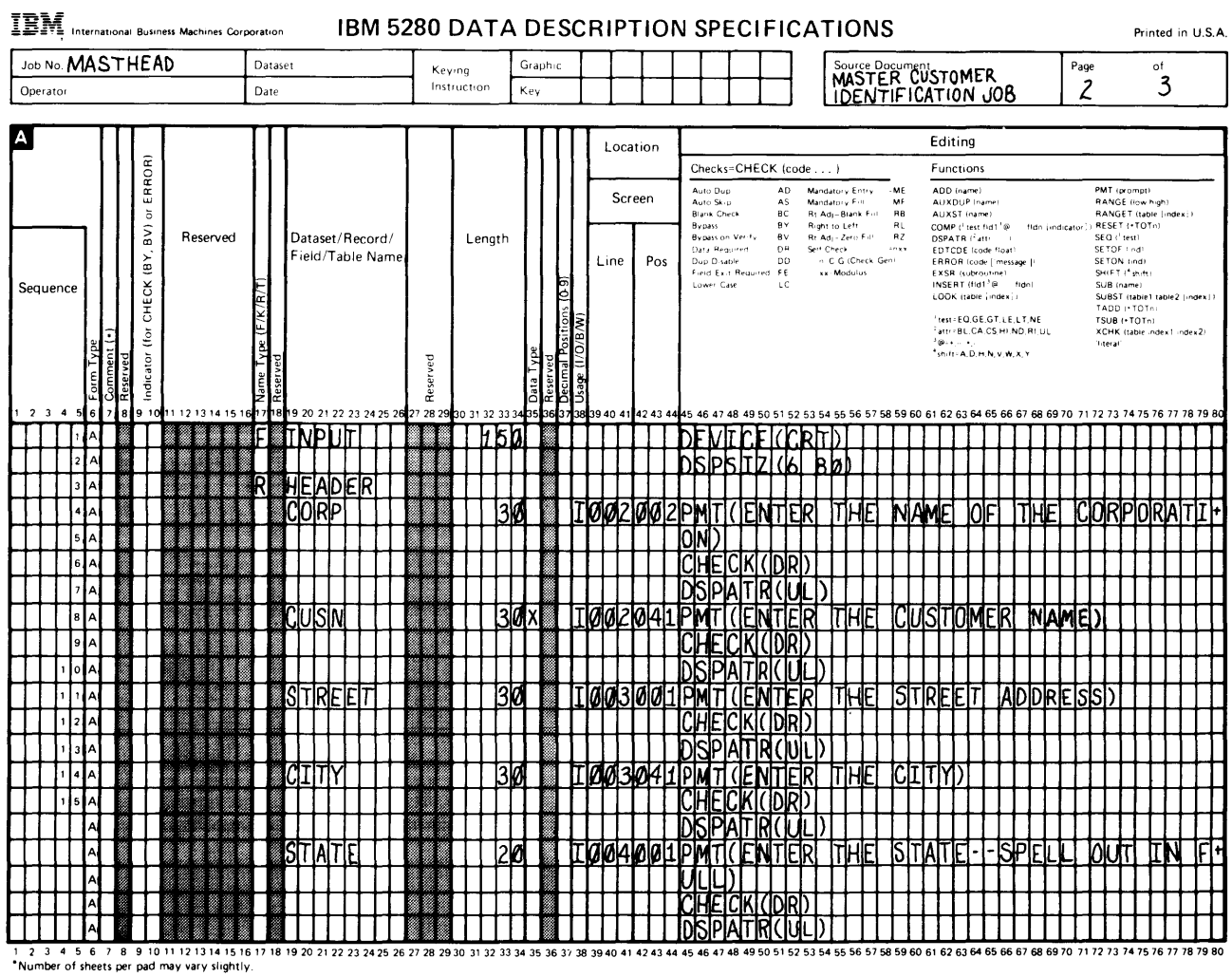
*Number of sheets per pad may vary slightly.

The record description is complete. The next description, the last one for the A specification for this example, is the description for the files.

DESCRIBING THE INPUT/OUTPUT DEVICE FILES ON THE A SPECIFICATION

On the first line on the A specification, place an F in column 17. The F identifies the line as a file description statement. Remember that Chapter 2 described two basic kinds of files: display and diskette. This line describes the display file. The name INPUT is arbitrary. You can use any name you wish for the display file. In column 45 write the keyword DEVICE followed by the parameter, CRT. The 150 in columns 32 through 34 tells the system that each record is to be 150 positions long. All records in a data set are the same length. Shorter records are padded with blanks to make them the specified length.

Next, you must tell the system what display size you are using. Write DSPSIZ for display size followed by the parameter (6 80). The (6 80) refers to 6 rows and 80 columns.



IBM International Business Machines Corporation — IBM 5280 DATA DESCRIPTION SPECIFICATIONS — Printed in U.S.A.

Job No. MASTHEAD	Dataset	Keying Instruction	Graphic								Source Document MASTER CUSTOMER IDENTIFICATION JOB	Page 2	of 3
Operator	Date		Key										

A

Sequence	Form Type	Comment (*)	Reserved	Indicator (for CHECK (BY, BV) or ERROR)	Reserved	Name Type (F/K/R/T)	Reserved	Dataset/Record/ Field/Table Name	Reserved	Length	Data Type	Decimal Positions (0-9)	Usage (I/O/B/W)	Location: Screen Line	Location: Screen Pos	Editing: Checks=CHECK (code . . .) / Functions	
1	A					F		INPUT		150						DEVICE(CRT)	
2	A															DSPSIZ(6 80)	
3	A					R		HEADER									
4	A							CORP		30			I	00	20	02	PMT(ENTER THE NAME OF THE CORPORATI+
5	A																ON)
6	A																CHECK(DR)
7	A																DSPATR(UL)
8	A							CUSN		30	X		I	00	20	41	PMT(ENTER THE CUSTOMER NAME)
9	A																CHECK(DR)
10	A																DSPATR(UL)
11	A							STREET		30			I	00	30	01	PMT(ENTER THE STREET ADDRESS)
12	A																CHECK(DR)
13	A																DSPATR(UL)
14	A							CITY		30			I	00	30	41	PMT(ENTER THE CITY)
15	A																CHECK(DR)
	A																DSPATR(UL)
	A							STATE		20			I	00	40	01	PMT(ENTER THE STATE--SPELL OUT IN F+
	A																ULL)
	A																CHECK(DR)
	A																DSPATR(UL)

*Number of sheets per pad may vary slightly.

One more file description is needed–the one that describes the diskette file. If this file description is not provided, no records can be written into the diskette data set. Go to the line on the A specification that follows the last field description and write the diskette file description. The name is MASTHEAD as defined by the instructions for the job (the length of 150 must be the same as the display file); the DEVICE parameter is DISK. The address for the diskette drive you are using must also be included as part of the parameter. You should be able to find this address on the outside of the diskette drive. For this example, the address is 4000.

Note: If you are not using drive 4000, substitute the number of the drive you are using wherever the drive number 4000 appears.

Look at the example if you have difficulty. Because the instructions tell you that the name for the data set is MASTHEAD, this is the name you must use for the diskette file description.

IBM International Business Machines Corporation **IBM 5280 DATA DESCRIPTION SPECIFICATIONS** Printed in U.S.A.

Job No MASTHEAD	Dataset	Keying Instruction	Graphic							Source Document MASTER CUSTOMER IDENTIFICATION JOB	Page 3	of 3
Operator	Date		Key									

A

Editing

Checks=CHECK (code . . .)

Auto Dup	=AD	Mandatory Entry	=ME
Auto Skip	=AS	Mandatory Fill	=MF
Blank Check	=BC	Rt Adj–Blank Fill	=RB
Bypass	=BY	Right to Left	=RL
Bypass on Verify	BV	Rt Adj–Zero Fill	RZ
Data Required	=DR	Self Check	=mxx
Dup Disable	DD	xx: C,G (Check Gen)	
Field Exit Required	=FE	xx: Modulus	
Lower Case	LC		

Functions

ADD (name)	PMT (prompt)
AUXDUP (name)	RANGE (low high)
AUXST (name)	RANGET (table [index])
COMP ([¹test fld1¹]@ fldn [indicator])	RESET (+TOTn)
DSPATR ([²attr . . .])	SEQ ([¹test)
EDTCDE (code [float]	SETOF (ind)
ERROR (code [message'])	SETON (ind)
EXSR (subroutine)	SHIFT ([⁴shift)
INSERT (fld1[¹]@ fldn)	SUB (name)
LOOK (table [index])	SUBST (table1 table2 [index])
	TADD (+TOTn)
¹test=EQ,GE,GT,LE,LT,NE	TSUB (+TOTn)
²attr=BL,CA,CS,HI,ND,RI,UL	XCHK (table index1 index2)
³@=+,–,*,/	'literal'
⁴shift=A,D,H,N,V,W,X,Y	

Sequence	Form Type	Comment (*)	Reserved	Indicator (for CHECK (BY, BV) or ERROR)	Reserved	Name Type (F/K/R/T)	Reserved	Dataset/Record/ Field/Table Name	Reserved	Length	Data Type	Reserved	Decimal Positions (0-9)	Usage (I/O/B/W)	Location: Screen Line	Location: Screen Pos	Editing
1	A														04	30	'CUSTOMER NUMBER:'
2	A							NUMBER		5				I	04	47	CHECK(DR BC)
3	A																DSPATR(UL)
4	A									4				I			CHECK(BY)
5	A							MARK		1				I	04	80	INSERT('H')
6	A					F		MASTHEAD		150							DEVICE(DISK X'4000')
7	A																
8	A																
9	A																
10	A																
11	A																
12	A																
13	A																
14	A																
15	A																
	A																
	A																
	A																
	A																
	A																

1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20 21 22 23 24 25 26 27 28 29 30 31 32 33 34 35 36 37 38 39 40 41 42 43 44 45 46 47 48 49 50 51 52 53 54 55 56 57 58 59 60 61 62 63 64 65 66 67 68 69 70 71 72 73 74 75 76 77 78 79 80

*Number of sheets per pad may vary slightly.

You have completed the descriptions for the files, record, and fields that are required for the first sample job. You have completed the A specification description and are now ready to define the job characteristics on the Z specification.

DESCRIBING THE JOB CHARACTERISTICS ON THE Z SPECIFICATION

The entries on the Z specification tell the system the characteristics of the job. This includes information such as:

- Job name
- Availability of the data set to other users
- Name of the diskette data set
- Order in which formats are to be used

The first thing that must be described on the Z specification is the job name and the name used for the data set on the diskette. On the first line of the Z specification, place a J in column 7.

IBM International Business Machines Corporation

IBM 5280 GENERAL UTILITY SPECIFICATIONS

Printed in U. S. A.

Job MASTER | Operator | Date | Keying Instruction | Graphic | Key

Description MASTER CUSTOMER IDENTIFICATION JOB | Page 1 of 3

Z

Sequence	Form Type	Job/Format/Subroutine: Name Type	Format ID (1-9, A0-Z9)	Name	Reserved	Repeat (1-9, N)	Mode (E/R)	AND(A)	Test Conditions: Position to be Tested (*POSnnnn)	Reserved	Condition	Character to Test for ('C')	Reserved	Next Format ID (0-9, A0-Z9)	Reserved	Options: Job Line / Entry Lines
1	Z	J		MASTER					*POS		EQ	'				TFILE (MASTHEAD 10)
2	Z								*POS		EQ	'				
3	Z								*POS		EQ	'				
4	Z								*POS		EQ	'				
5	Z								*POS		EQ	'				
6	Z								*POS		EQ	'				
7	Z								*POS		EQ	'				
8	Z								*POS		EQ	'				
9	Z								*POS		EQ	'				
10	Z								*POS		EQ	'				
11	Z								*POS		EQ	'				
12	Z								*POS		EQ	'				
13	Z								*POS		EQ	'				
14	Z								*POS		EQ	'				
15	Z								*POS		EQ	'				
	Z								*POS		EQ	'				
	Z								*POS		EQ	'				
	Z								*POS		EQ	'				
	Z								*POS		EQ	'				
	Z								*POS		EQ	'				

Job/Format/Subroutine: Name; Test Conditions: Position to be Tested (*POSnnnn); Options: Job Line — CFILE (data set); DATE (+DMY +YMD); EDITC ([cuptd]); ENTRATR (attr); EXITATR (attr); JOBOPT ([+NOPMT][+NOOPEN]); PRTFILE (data set); SHARE (names); SHARER (names); STATUS (name); TFILE (data set [deltreq]); attr: BLCS HI ND RI UL; Entry Lines — CLRL (number); EOJ ([job dev [+PASS][!]]; SLNO (line); WRITE (name)

1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20 21 22 23 24 25 26 27 28 29 30 31 32 33 34 35 36 37 38 39 40 41 42 43 44 45 46 47 48 49 50 51 52 53 54 55 56 57 58 59 60 61 62 63 64 65 66 67 68 69 70 71 72 73 74 75 76 77 78 79 80

*Number of sheets per pad may vary slightly.

The J tells DE/RPG that this is a job description statement. Place the name of the job in columns 10 through 17; it can be no longer than 8 positions. The instructions tell you to use the name MASTER. Beginning in column 55, write the keyword TFILE followed by the parameter (MASTHEAD). You should recognize MASTHEAD as being the name of the diskette data set you provided in the diskette file statement on the A specification. The name that you specify for the diskette data set on the A specification must always be the same as the name you specify for the parameter of the TFILE keyword on the Z specification. Now write the number 10 as part of the parameter: (MASTHEAD 10). The 10 tells DE/RPG to insert a deleted record after every 10 records that are written to the data set. You determine the frequency of deleted records. These deleted records are optional and do not have to be specified; however, they make the insertion of future records in the data set faster.

The job description is complete, but the format description on the Z specification remains to be defined.

DESCRIBING THE FORMATS ON THE Z SPECIFICATION

Go to the line following the job description statement. Place the format identifier (ID) X1 in columns 8 and 9. This ID was provided in the supervisor's instructions. The characters X1 identify the format. Next, write the format name starting in column 10. Notice that the name of the format is the same as the name of the record on the A specification. Because the program has only one record, there is only one format description. Next, place an E in column 21. The E stands for entry format. Two types of formats can be defined on the Z specification: the entry formats and review formats. The entry format information tells DE/RPG to use the field definitions of the matching display record name to display information during the enter mode.

Finally, define the review format. Place an R rather than an E in column 21. Place the format identification (ID) in columns 45 and 46. This tells the system to use the record that matches the format ID to display information for the update, verify, and rerun modes. The format ID in columns 45 and 46 refers back to an entry format which in turn points to a record described for the CRT file on the A specification. In this example, the ID is X1. The format name for ID X1 is HEADER which is also the name of a display record on the A specification.

You have completed the job definition for the master customer file job. Your first program is complete.

IBM International Business Machines Corporation

IBM 5280 GENERAL UTILITY SPECIFICATIONS

Printed in U. S. A.

Job MASTER | Keying Instruction | Graphic Key | Description MASTER CUSTOMER IDENTIFICATION JOB | Page 1 of 1

Operator | Date

Z

Sequence	Form Type	Name Type	Job/Format/Subroutine: Format ID (1-9, A0-Z9)	Job/Format/Subroutine: Name	Reserved	Repeat (1-9, N)	Mode (E/R)	AND(A)	Test Conditions: Position to be Tested (*POSnnnn)	Test Conditions: Reserved	Test Conditions: Condition	Test Conditions: Character to Test for ('C')	Reserved	Next Format ID (0-9, A0-Z9)	Reserved	Options: Job Line / Entry Lines
1	Z	J		MASTER					*POS		EQ	' '				TFILE (MASTHEAD 10)
2	Z								*POS		EQ	' '				
3	Z	X	1	HEADER			E		*POS		EQ	' '				
4	Z						R		*POS		EQ	' '		X1		
5	Z								*POS		EQ	' '				
6	Z								*POS		EQ	' '				
7	Z								*POS		EQ	' '				
8	Z								*POS		EQ	' '				
9	Z								*POS		EQ	' '				
10	Z								*POS		EQ	' '				
11	Z								*POS		EQ	' '				
12	Z								*POS		EQ	' '				
13	Z								*POS		EQ	' '				
14	Z								*POS		EQ	' '				
15	Z								*POS		EQ	' '				
	Z								*POS		EQ	' '				
	Z								*POS		EQ	' '				
	Z								*POS		EQ	' '				
	Z								*POS		EQ	' '				
	Z								*POS		EQ	' '				

Job Line: CFILE (data set) | DATE (*DMY/*YMD) | EDITC (|cuptd|) | ENTRATR (attr...) | EXITATR (attr...) | JOBOPT (|*NOPMT|*NOOPEN|) | PRTFILE (data set) | SHARE (names) | SHARER (names) | STATUS (name) | TFILE (data set (delfreq)) | attr: BL,CS,HI,ND,RI,UL

Entry Lines: CLRL (number) | EOJ (|job dev (*PASS)|) | SLNO (line) | WRITE (name)

*Number of sheets per pad may vary slightly.

Only two tasks remain before you can use the program to enter customer identification information. The first task is entering the program into the 5280 system. The source entry program is available to help you do this. Chapter 5 describes how to use the source entry program. Secondly, you must compile the program once it is in the system. Chapter 5 also describes this process. Once the program has been entered and compiled, it is ready to be used for data entry. Chapter 6 shows you how to use the program to enter data.

SUMMARY OF CHAPTER 4

You have completed the descriptions of the job, formats, files, record, and fields on the Z and A specifications. Before you proceed, however, try to answer the following questions.

1. Place an A by the items that the A specification describes and a Z by those that the Z specification describes:

   a. jobs _____

   b. records _____

   c. files _____

   d. data sets _____

   e. fields _____

   f. formats _____

   g. prompts _____

   h. literals _____

2. Two files are described in this program. Can you name these files and the device that they refer to:

   a. _____

   b. _____

3. Suppose you are using a larger display and you want to describe a field that is located in row 8, column 39. What would be your entries in columns 39 through 44 of the A specification?

   _____

4. Which one of the following definitions matches the function of CHECK(DR)? _____

   a. Right-adjust with blank fill

   b. Pad the field with blanks

   c. Nonblank entry required

5. Where are the identical names required:

   a. Records and formats _____

   b. Files and formats _____

   c. Diskette files and transaction files _____

   d. CRT files and jobs _____

   e. Files and records _____

6. What does the R on the format line on the Z specification mean?

   _____

Check your answers against the answers provided in Appendix A. If you have not been able to answer the questions, begin at Chapter 2 and read through the text again.

Chapter 5. Entering and Compiling the Program for the Master Customer Identification Job

Review the process involved in using DE/RPG as shown in Figure 2 in Chapter 3. You are now ready to begin Step 2.

This chapter will teach you how to take the information you have gathered on the Z and A specifications and enter it into the 5280 system so it can be compiled and used.

Assume that you will be using a system which consists of 32 K bytes of main storage, a single display, and two diskette drives (no printer).

If you are going to use your keyboard/display data station to enter the program, you will have to perform the following procedures:

1. Configure the system with the appropriate partition sizes for the source entry program (16 K) and the compiler (9 K). (Generally, this system setup will have been done by your supervisor prior to your using the system.)

2. Perform the IPL.

   Note: The IPL must include either SYSCFA or SYSHELP.

3. Load and use the source entry program to enter the statement descriptions from the Z and A specifications.

4. Load the DE/RPG compiler and describe the conditions of the compilation.

CONFIGURING YOUR SYSTEM AND PERFORMING THE IPL

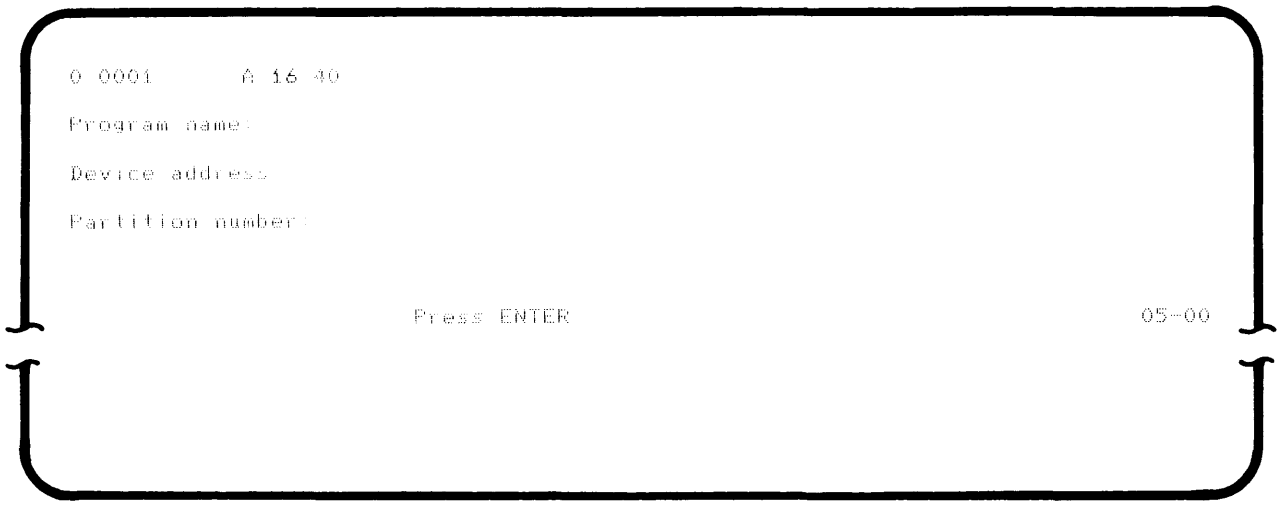
The *System Control Programming Reference/Operation Manual* provides detailed procedures for performing the IPL and system configuration. You will need at least one foreground partition that is 13 K bytes.

Once the system has been configured to fit your requirements and has completed IPL, you are ready to use the source entry program.

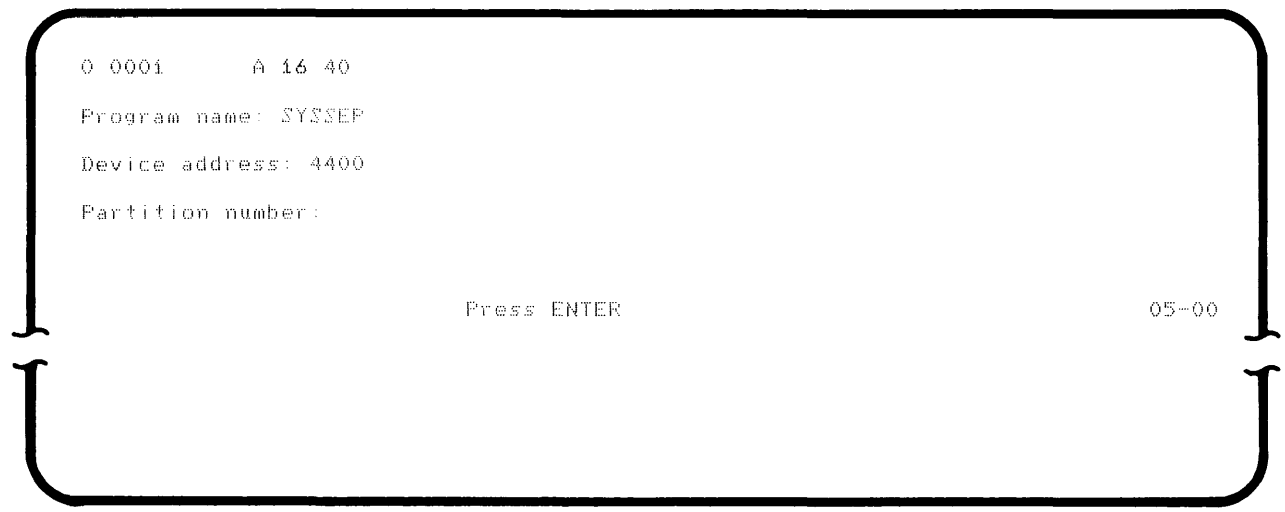
If an error should occur, refer to the *IBM 5280 Message Manual* for an explanation of the error code and suggested response.

USING THE SOURCE ENTRY PROGRAM

Immediately after the IPL has been completed, this display appears.

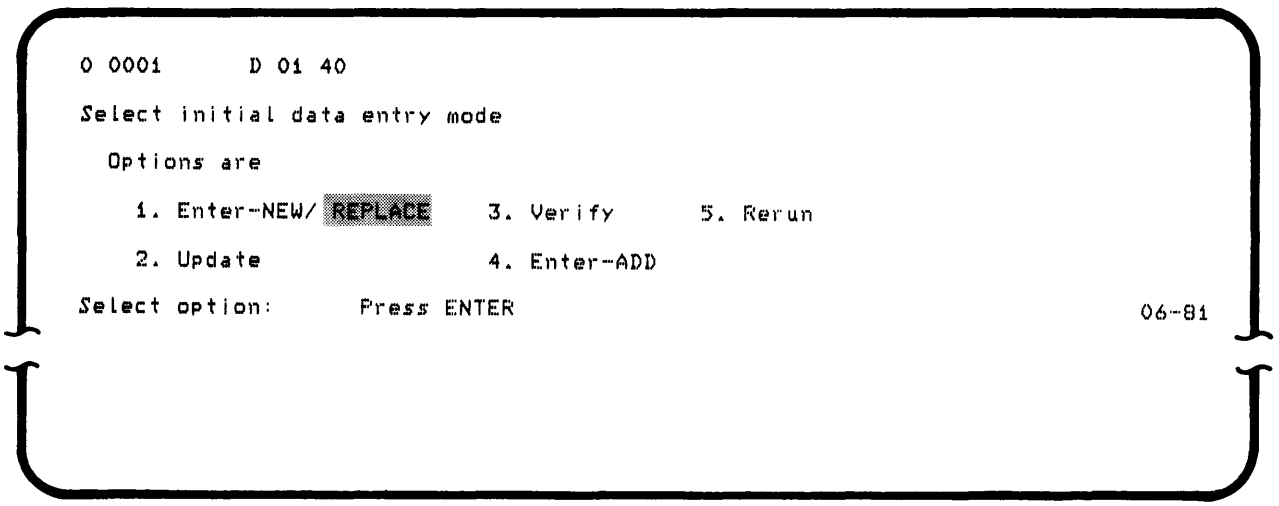


Place the diskette containing the source entry program in drive 4400. If you are not using drive 4400, substitute the number for the drive you are using wherever drive number 4400 appears in the following samples.



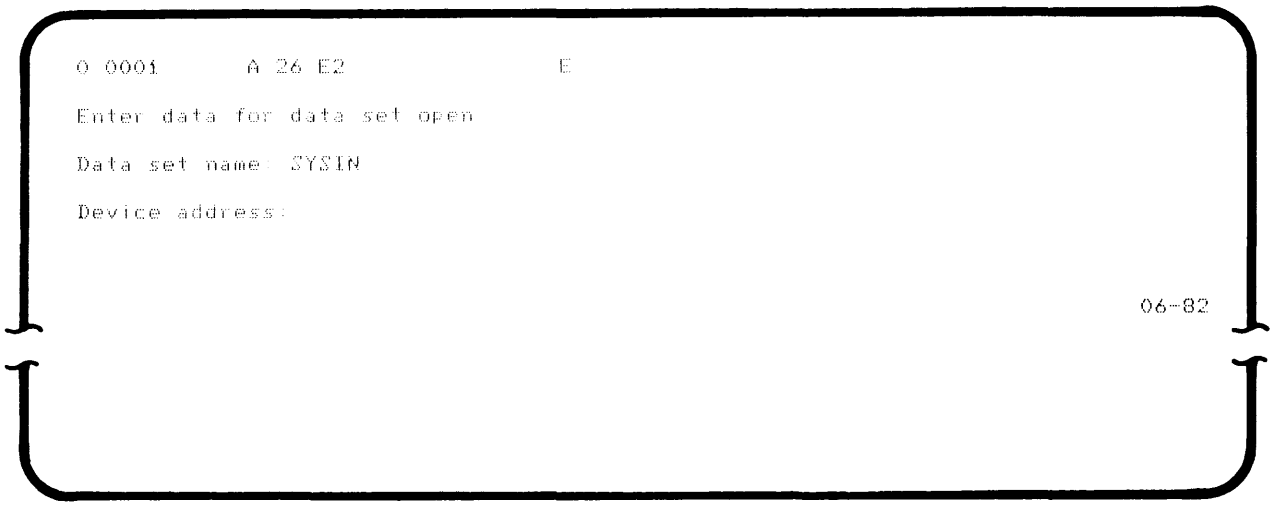
The responses you should provide are indicated in color on the sample display. SYSSEP is the name by which the 5280 recognizes the source entry program.

When you have completed the entries shown, press the Enter key. The following display appears.

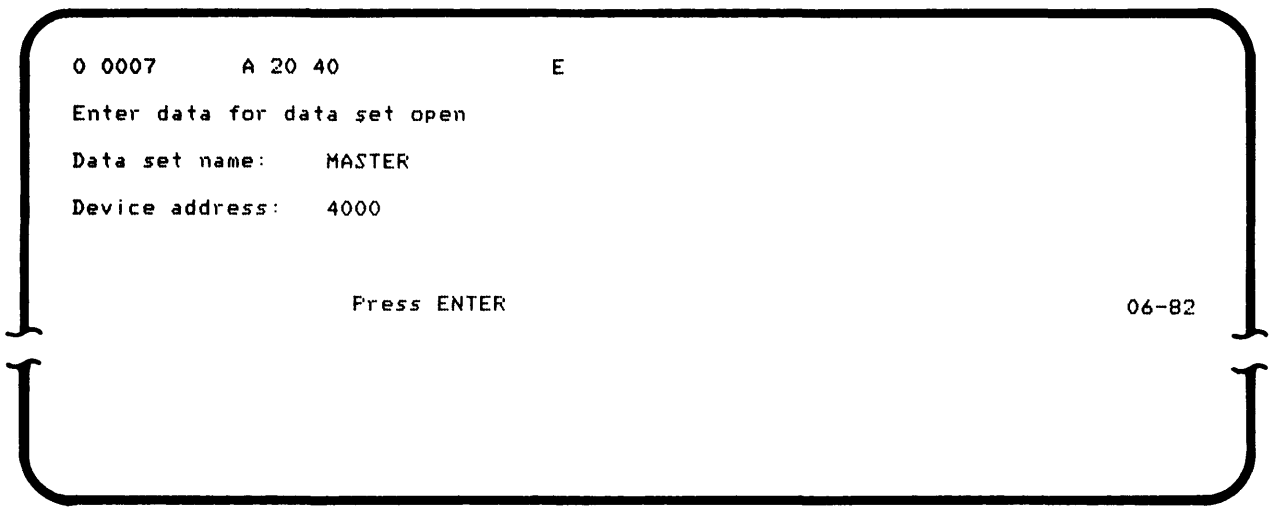


This is the first time you have used the source entry program for entering this program, so select the option that says Enter (New). (The only time you select option 1 is for the first time you are entering the source DE/RPG program; if you use this option any other time, it will erase everything that you had previously put in the data set.)

When you press the Enter key, a display containing a default data set name appears



Change the data set name to the name of your program (MASTER). Change the drive to 4000 because this is the number of the drive that contains the diskette on which your MASTER data set will be located. The MASTER data set will be your DE/RPG source program. Be sure to place a diskette in drive 4000.



Press the Enter key. If the data set name specified is not on the diskette, a display appears requesting you to specify either a retry of the name and device entry or the allocation of the data set. (If any other error occurs, prompt 06-82 is displayed again.)

```
0 0001      D 01 40                E

Data set open failed

  Options are

    1. Retry

    2. Allocate

Select option:      Press ENTER                                    06-83
```

Select option 2. Because the diskette in drive 4000 does not contain a data set named MASTER, you must now allocate the data set as follows. Press the Enter key and the allocation display appears.

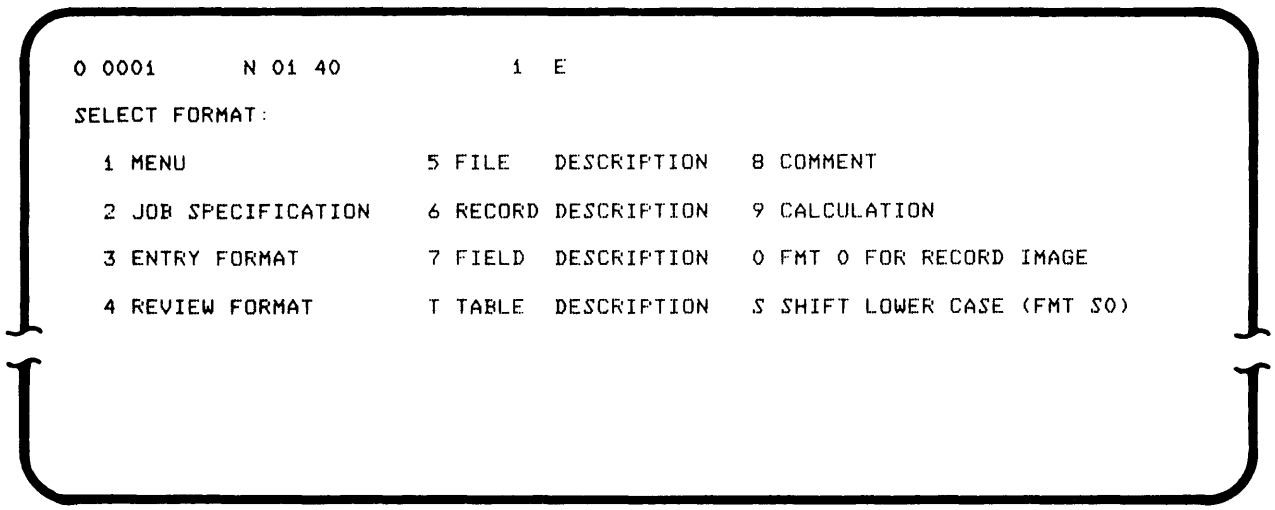
```
0 0001      A 26 D4                E

Enter data for data set allocate

Data set name:     MASTER

Device address:    4000    Owner ID:

Exchange type:     I  Number of records:    000200

                      Press ENTER                                 06-84
```

Enter the number 50 as the number of records entry and use a field exit key to leave the field. Press the Enter key.

You have now allocated a data set named MASTER on the diskette in drive 4000. The name you provide for this data set can be any acceptable name you wish as long as you remember that name and specify it as the source name during the compilation. You are specifying that the data set consists of 50 records. The data set containing the DE/RPG source program must be at least as large as the actual program you are entering. It should be larger than the program to provide space for additions or changes to the program.

After choosing option 2, the job specification prompt is displayed. The following steps indicate the process involved in entering your program.

When you press the Enter key, the menu for the source entry program appears.



A menu is a programming name given to this type of display that allows you to select from a variety of options.

Before you can understand which option to select, you must understand the sequence in which the contents of the Z and A specifications must be entered. The normal sequence for entering programs into the 5280 for the source entry program is:

A. Job statement (Z specification)

B. Entry format statement (Z specification)

C. Review format statement (Z specification)

D. Display file statement (A specification)

E. First record statement (A specification)

F. Associated field statements (A specification)

G. Next record statement and its associated field statements (A specification)

H. Diskette file statement for the data set being created (A specification)

I. Diskette file statements for other data sets used by the program (tables, and so on) (A specification)

Review the contents of your Z and A specifications before beginning.

Select option 2 from the menu and press the Enter key to begin the process of entering your program.

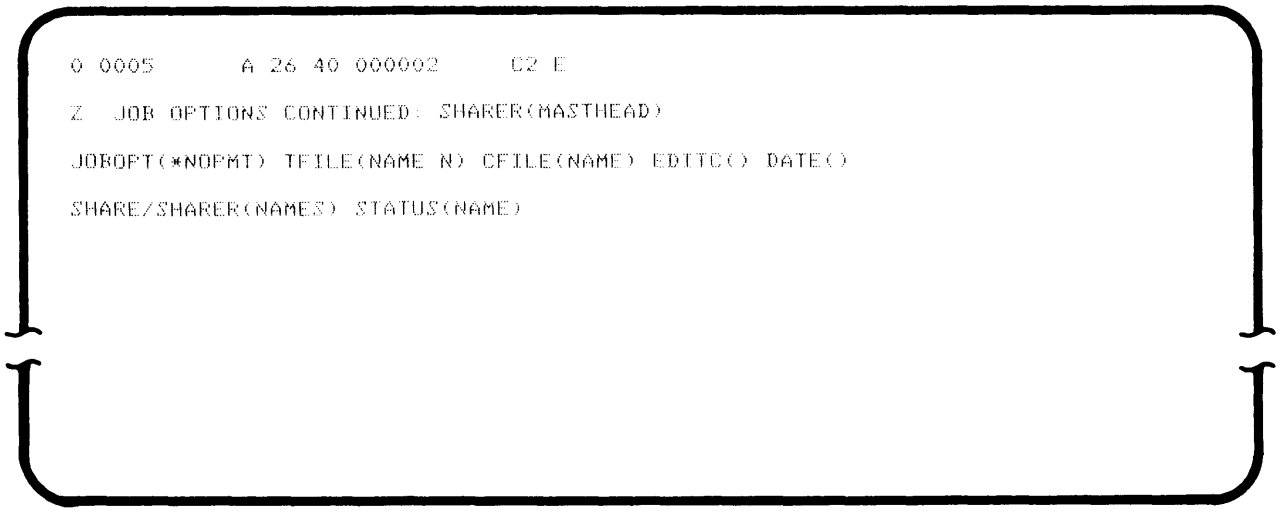
After choosing option 2, the job specification prompt is displayed. The following steps indicate the process involved in entering your program.

1. Job specification prompt

```
0 0073      A 08 40 000001     2  E
Z  JOB SPECIFICATION
   Name: MASTER
Options: TFILE(MASTHEAD 10)
JOBOPT(*NOPMT) TFILE(name n) CFILE(name) EDITC() DATE()
SHARE/SHARER(names) STATUS(name) PRTFILE(name) ENTRATR() EXITATR()
```

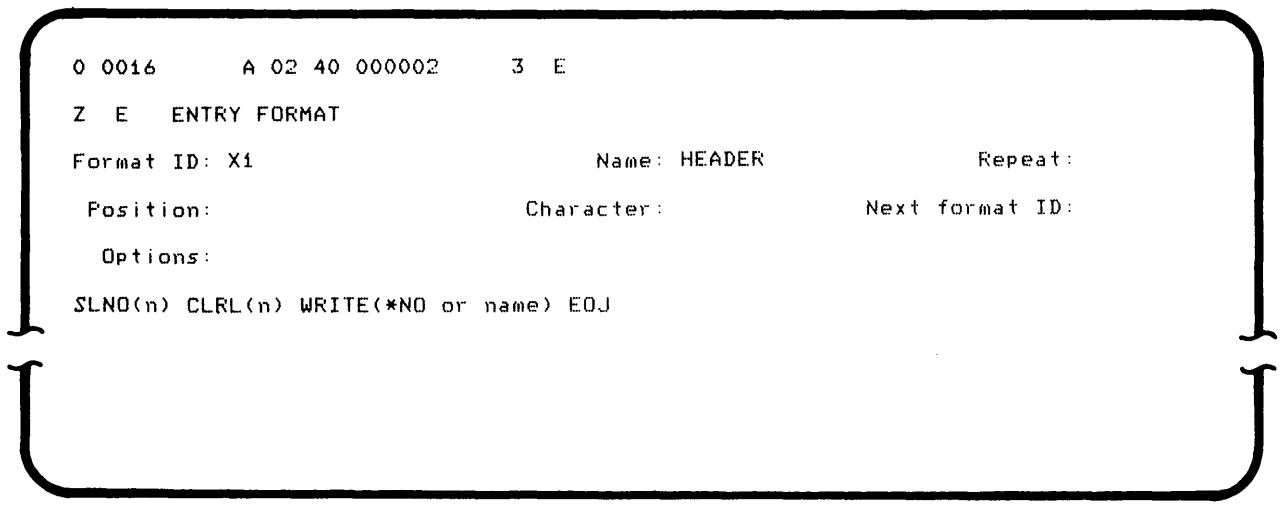
Make the entries indicated in color. Press the Enter key, and the job options continued prompt is automatically displayed.

2. Job options continued prompt



Make no entries, just press the Next Fmt key. The entry format prompt is displayed.

3. Entry format prompt



Make the entries shown and press the Enter key. The entry format prompt is displayed again. Make no more entries, just press the Next Fmt key. The review format prompt is displayed.

4. Review format prompt

```
0 0000      Y 00 F1 000003     4  E
Z  REVIEW FORMAT
        And(A):
      Position:
     Character:
Next format ID: X1
```

Make the entries as shown and press the Enter key. The review format prompt is displayed again. Now press the Next Fmt key to display the menu again.

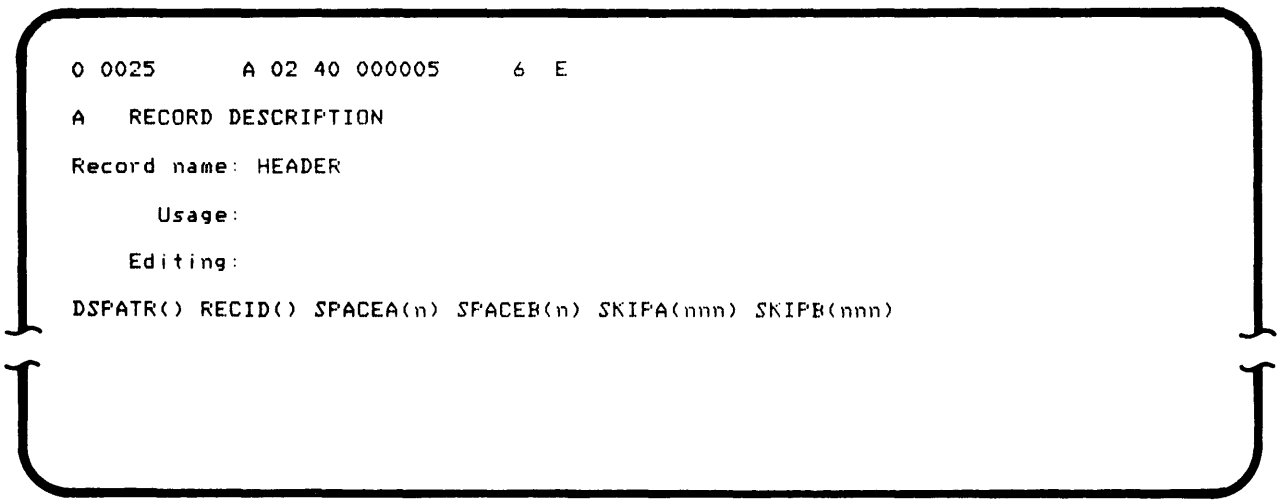
Select option 5, and press the Enter key. The file description prompt is displayed.

5. CRT file description prompt

```
0 0069      A 12 40 000004     5  E
A   FILE DESCRIPTION
File name: INPUT
   Length:   150
    Usage:          Editing: DEVICE(CRT) DSPSIZ(6 80)
BLKING() DEVICE() LABEL() FORM() NUMENT() DSPSIZ() LOGON() INDEX() MARK/VMARK()
```

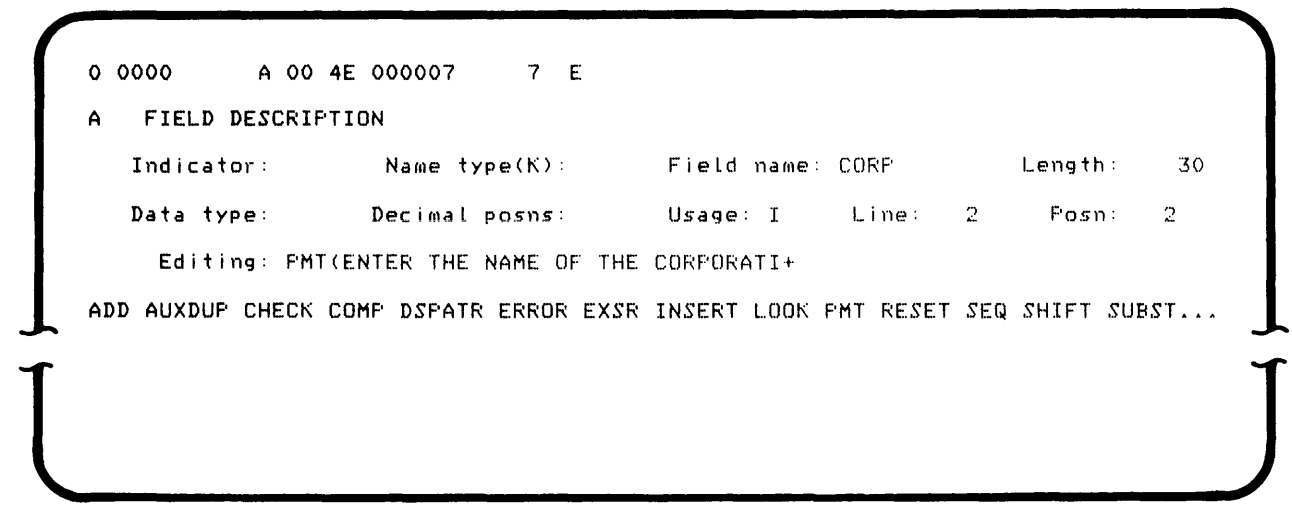
Make the entries as shown. Although DSPSIZ appeared on a separate line on the A specification sheet, it is correct to enter it as shown if there is room. Press the Enter key and the file description prompt is displayed again. Press the Next Fmt key to display the record description prompt.

6. Record description prompt



Make the entry shown. Press the Enter key and the field description prompt is displayed.

7. CORP field description prompt.



Make the entries shown, including the + continuation character. For all the field descriptions, use the Field Exit key to leave the length, line, and position fields. A + continuation character in the last position of the editing field indicates that there is more information to be entered.

When the field description prompt is displayed, you can:

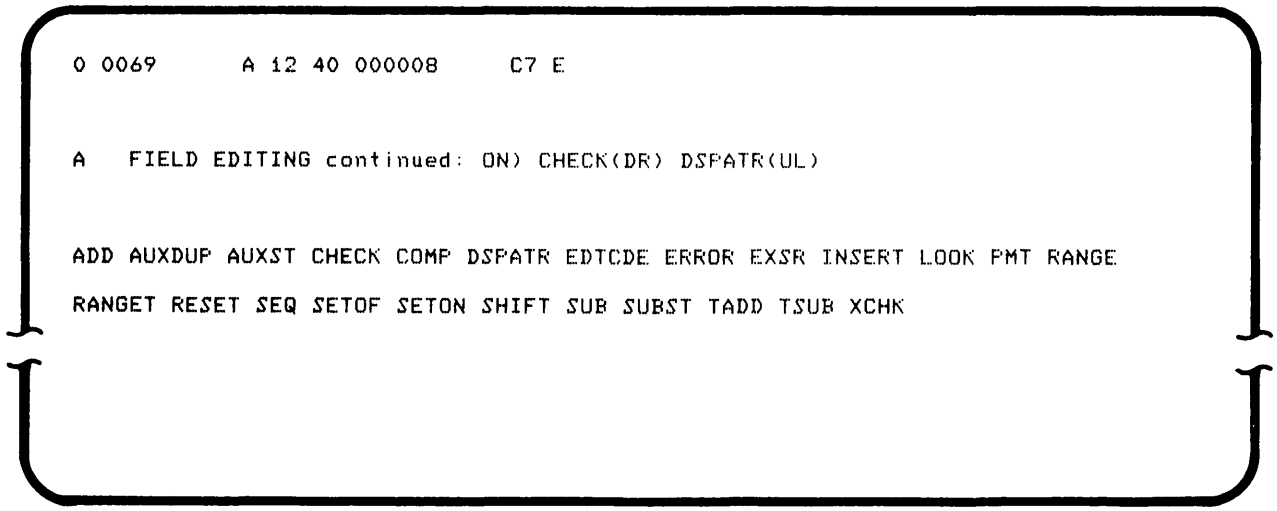
- Advance to the editing field of this prompt by pressing the Field Exit key several times. Then enter the remaining data and press the Enter key. The field description prompt is displayed again; you can now enter data for another field.

- Press the Next Fmt key. The field editing continued prompt is displayed (as in step 8).

- Select a continuation prompt by pressing the Sel Fmt key. Then press C (alpha shift) 7, for the prompt ID number. The field editing continued prompt is displayed (as in step 8).

Note: When running SYSSEP, you can select another prompt at any time by pressing the Sel Fmt key; then press the prompt ID number.

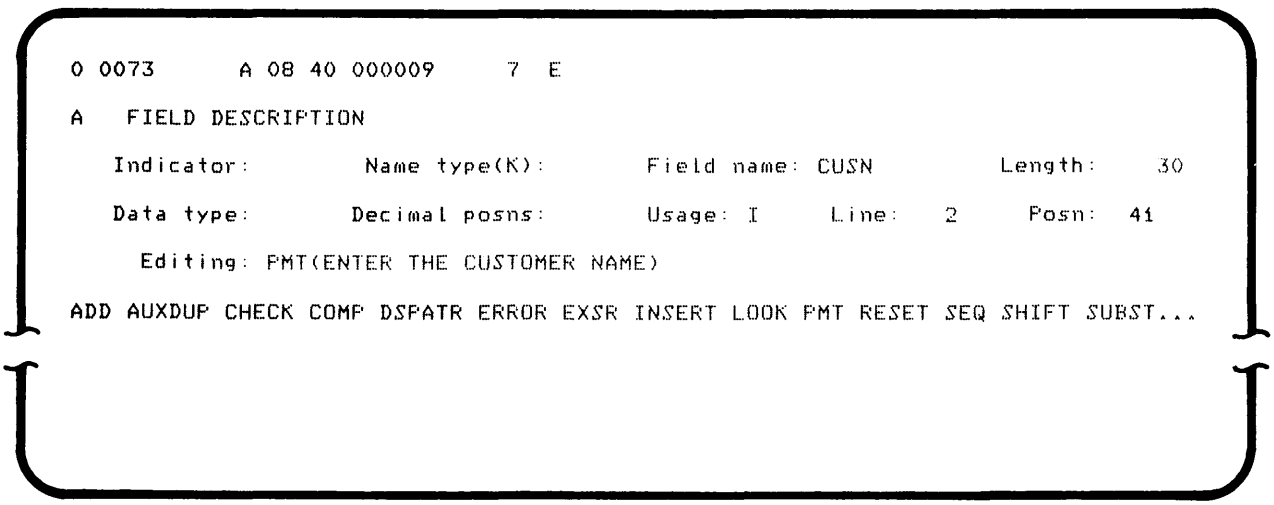
8. CORP field editing continued prompt



This prompt is displayed only if you requested a continuation prompt during the previous step. Make the entries shown, then press the Enter key. The field editing continued prompt is displayed again. Press the Next Fmt key to display a field description prompt.

Note: Although items such as CHECK(DR) and DSPATR(UL) were listed on separate lines on the A specification sheet, it is correct to enter them on the same prompt if there is enough room in the editing field.

9. CUSN field description prompt



Make the entries shown, then press the Enter key. The field description prompt is displayed again. You can continue entering data in the field description prompt, or you can request a field editing continued prompt as outlined in step 7.

10. CUSN field editing continued prompt

```
0 0065      A 16 40 000010      C7 E

A   FIELD EDITING continued: CHECK(DR) DSPATR(UL)

ADD AUXDUP AUXST CHECK COMP DSPATR EDTCDE ERROR EXSR INSERT LOOK PMT RANGE
RANGET RESET SEQ SETOF SETON SHIFT SUB SUBST TADD TSUB XCHK
```

This prompt is displayed only if you requested a continuation prompt during the previous step. Make the entries shown, then press the Enter key. The field editing continued prompt is displayed again. Press the Next Fmt key to display a field description prompt.

11. STREET field description prompt

```
0 0074      A 07 40 000011      7  E

A   FIELD DESCRIPTION

   Indicator:          Name type(K):        Field name: STREET        Length:     30

   Data type:        Decimal posns:         Usage: I      Line:    3     Posn:    1

     Editing: PMT(ENTER THE STREET ADDRESS)

ADD AUXDUP CHECK COMP DSPATR ERROR EXSR INSERT LOOK PMT RESET SEQ SHIFT SUBST...
```

Make the entries as shown, including a continuation character in the last position of the editing field. Press the Enter key. The field description prompt is displayed again. You can continue entering data in the field description prompt, or you can request a field editing continued prompt as outlined in step 7.

12. STREET field editing continued prompt

```
0 0065      A 16 40 000012      C7 E

A   FIELD EDITING continued: CHECK(DR) DSPATR(UL)

ADD AUXDUP AUXST CHECK COMP DSPATR EDTCDE ERROR EXSR INSERT LOOK PMT RANGE
RANGET RESET SEQ SETOF SETON SHIFT SUB SUBST TADD TSUB XCHK
```

This prompt is displayed only if you requested a continuation prompt during the previous step. Make the entries as shown, then press the Enter key. The field editing continued prompt is displayed again. Press the Next Fmt key to display a field description prompt.

13. CITY field description prompt

```
0 0064      A 17 40 000014      7  E
A   FIELD DESCRIPTION
   Indicator:        Name type(K):       Field name: CITY         Length:    30
   Data type:       Decimal posns:       Usage: I     Line:   3     Posn:   41
     Editing: PMT(ENTER THE CITY)
ADD AUXDUP CHECK COMP DSPATR ERROR EXSR INSERT LOOK PMT RESET SEQ SHIFT SUBST...
```

Make the entries as shown, and press the Enter key. The field description prompt is displayed again. You can continue entering data in the field description prompt, or you can request a continuation prompt as outlined in step 7.

14. CITY field editing continued prompt

```
0 0065      A 16 40 000015     C7 E

A   FIELD EDITING continued: CHECK(DR) DSPATR(UL)

ADD AUXDUP AUXST CHECK COMP DSPATR EDTCDE ERROR EXSR INSERT LOOK PMT RANGE
RANGET RESET SEQ SETOF SETON SHIFT SUB SUBST TADD TSUB XCHK
```

This prompt is displayed only if you requested a continuation prompt during the previous step. Make the entries as shown, then press the Enter key. The field editing continued prompt is displayed again. Press the Next Fmt key to display a field description prompt.

15. STATE field description prompt

```
0 0000      A 00 60 000016     7  E
A   FIELD DESCRIPTION
    Indicator:        Name type(K):       Field name: STATE        Length:    20
    Data type:        Decimal posns:      Usage: I     Line:   4    Posn:   1
      Editing: PMT(ENTER THE STATE -- SPELL OUT IN-
ADD AUXDUP CHECK COMP DSPATR ERROR EXSR INSERT LOOK PMT RESET SEQ SHIFT SUBST...
```

Notice that the continuation character is a minus sign. The minus sign is used in this example because the next character to be entered in the editing field is the blank between 'In' and 'Full'. A + continuation character does not allow a leading blank in a continued editing field. By using a - continuation character, you can continue with the blank in the first position of the next prompt.

Make the entries as shown, then press the Enter key. The field description prompt is displayed again. You can continue entering data in the field description prompt, or you can request a field editing continued prompt as outlined in step 7.

16. STATE field editing continued prompt

```
0 0072      A 09 40 000017     C7 E

A   FIELD EDITING continued:  FULL) CHECK(DR) DSPATR(UL)

ADD AUXDUP AUXST CHECK COMP DSPATR EDTCDE ERROR EXSR INSERT LOOK PMT RANGE
RANGET RESET SEQ SETOF SETON SHIFT SUB SUBST TADD TSUB XCHK
```

This prompt is displayed only if you requested a continuation prompt during the previous step. Make the entries as shown, then press the Enter key. The field editing continued prompt is displayed again. Press the Next Fmt key to display a field description prompt.

17. CUSTOMER NUMBER literal field description prompt

```
0 0063      A 18 40 000018     7  E
A   FIELD DESCRIPTION
   Indicator:        Name type(K):       Field name:              Length:
   Data type:       Decimal posns:       Usage: O     Line:   4     Posn:  30
     Editing: 'CUSTOMER NUMBER:'
ADD AUXDUP CHECK COMP DSPATR ERROR EXSR INSERT LOOK PMT RESET SEQ SHIFT SUBST...
```

Make the entries as shown, then press the Enter key. The next field description prompt is displayed.

18. Number field description prompt

```
0 0068      A 13 40 000020     7  E
A   FIELD DESCRIPTION
   Indicator:         Name type(K):       Field name: NUMBER       Length:      5
   Data type:       Decimal posns:        Usage: I     Line:   4     Posn:  47
     Editing: CHECK(DR BC) DSPATR(UL)
ADD AUXDUP CHECK COMP DSPATR ERROR EXSR INSERT LOOK PMT RESET SEQ SHIFT SUBST...
```

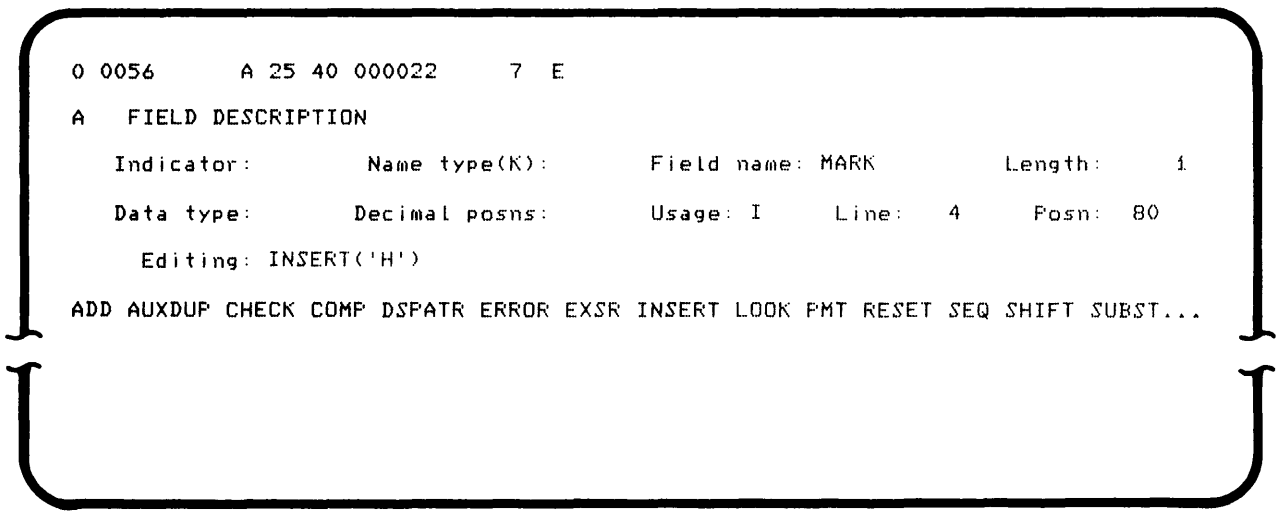
Make the entries as shown, then press the Enter key. The next field description prompt is displayed.

19. Bypass field description prompt

```
0 0054      A 27 40 000021     7  E
A   FIELD DESCRIPTION
   Indicator:         Name type(K):       Field name:              Length:      4
   Data type:       Decimal posns:        Usage: I     Line:          Posn:
     Editing: CHECK(BY)
ADD AUXDUP CHECK COMP DSPATR ERROR EXSR INSERT LOOK PMT RESET SEQ SHIFT SUBST...
```

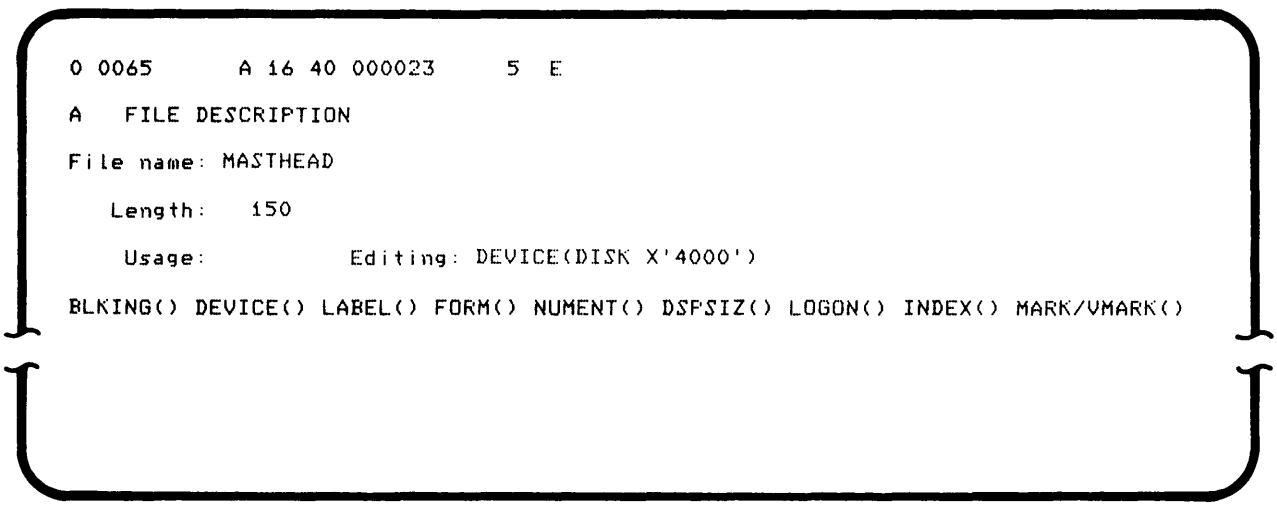
Make the entries as shown, then press the Enter key. The next field description prompt is displayed.

20. MARK field description prompt



Make the entries as shown, then press the Enter key. The field description prompt is displayed again. Because you have finished entering field descriptions, use the Sel Fmt 5 key sequence to display the file description prompt.

21. Diskette file description prompt



Make sure a file description prompt is displayed. Make the entries as shown, then press the Enter key. The prompt is displayed again. Use the End of Job key sequence to end the process of entering the program.

The following prompt appears.

```
0 0001      D 01 40 000024     5  E
End of job.    Do you want to write statistics?
  Options are
    1. Yes
    2. No
Select option:      Press ENTER                                    06-89
```

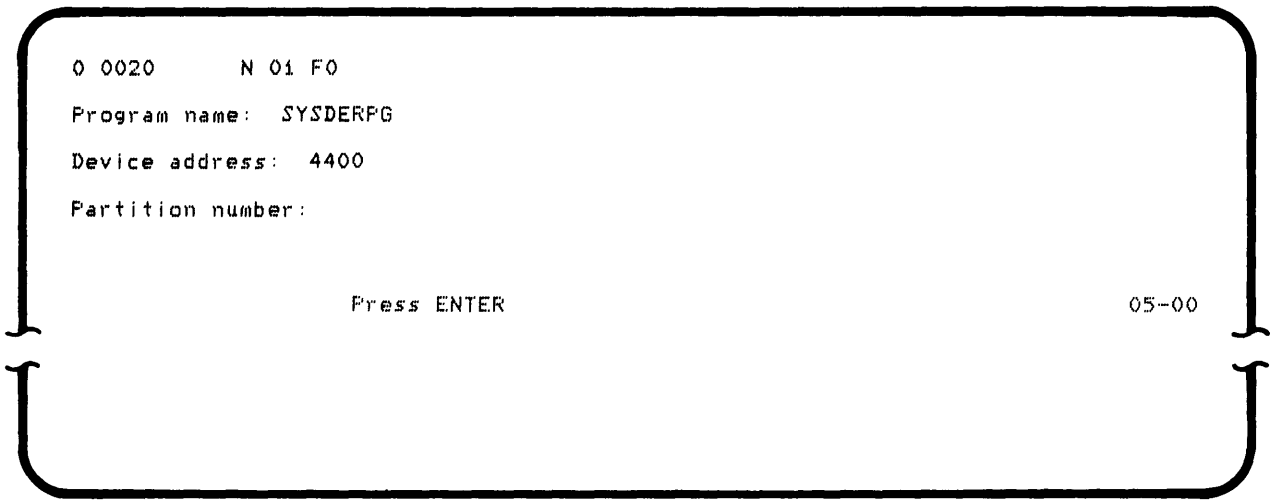
Select option 2 as it is not necessary for you to have production statistics posted. Press the Enter key, and the program load prompt appears again.

```
0 0001      A 16 40
Program name:
Device address:
Partition number:

                          Press ENTER                              05-00
```

Once you have entered your program using the source entry program, you are ready to use the DE/RPG compiler to change it into a machine-readable form.

USING THE DE/RPG COMPILER

Be sure that the diskette containing the DE/RPG compiler is in diskette drive 4400 (or the drive you have been using instead of 4400). Respond to the prompt that appeared at the conclusion of the source entry program in the following way.



Press the Enter key and the following display appears.

```
0 0001      A 16 E2

                          DE/RPG COMPILER

Enter the following information for      Source file.

Data set name:     SYSIN

Device address:    4000

                          Press ENTER                                  12-01
```

The values that appear on the display are the default values. Change these as shown below.

```
0 0007      A 10 40

                          DE/RPG COMPILER

Enter the following information for      Source file.

Data set name:     MASTER

Device address:    4000

                          Press ENTER                                  12-01
```

The name of the source file must be the name you gave the source DE/RPG program which you used the source entry program to create. For this example, the name is MASTER. The drive is 4000 because you will be using the diskette that has remained in drive 4000 (or its equivalent for your system). Press the Enter key.

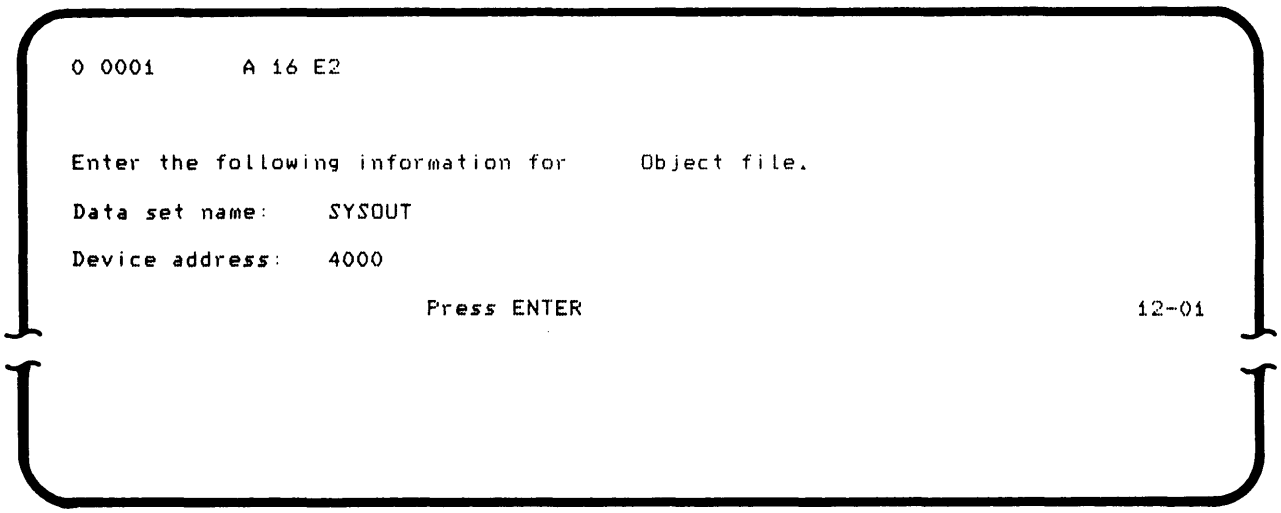
The next two displays that appear are for work files. Accept both these displays as provided.

```
0 0001      A 16 E2
Enter the following information for    Work file 1.
Data set name:   SYSUT001
Device address:  4400
                          Press ENTER                                12-01
```

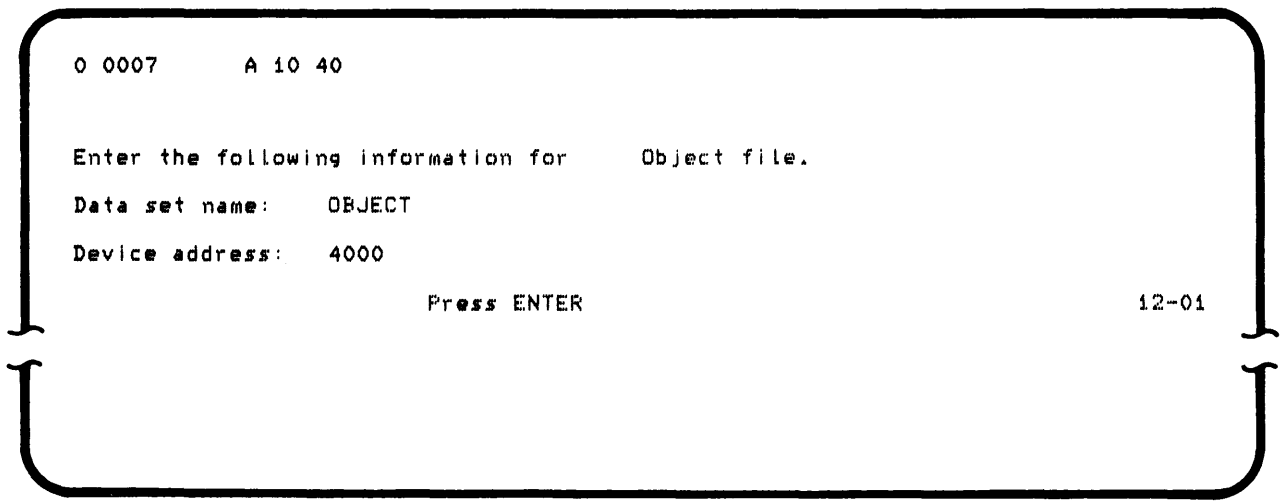
Press the Enter key to continue.

```
0 0001      A 16 E2
Enter the following information for    Work file 2.
Data set name: SYSUT002
Device address: 4400

                          Press ENTER                                12-01
```

Press the Enter key and the display for the object program data set appears.



Change the entries as shown below.



The data set name can be any name you choose. It will be the name that you use to load the program in the future. In this example, the name OBJECT is used. The device address tells the system where to put the data set. You will be placing this data set on the same diskette that contains the source program (MASTER), which is in drive 4000 (or its equivalent for your system).

Press the Enter key and the following display appears.

```
0 0001      D 01 40
Data set   OBJECT          not found on device  4000
Options are
  1. Reenter name and device
  2. Allocate space for data set
Select option:      Press ENTER                              12-02
```

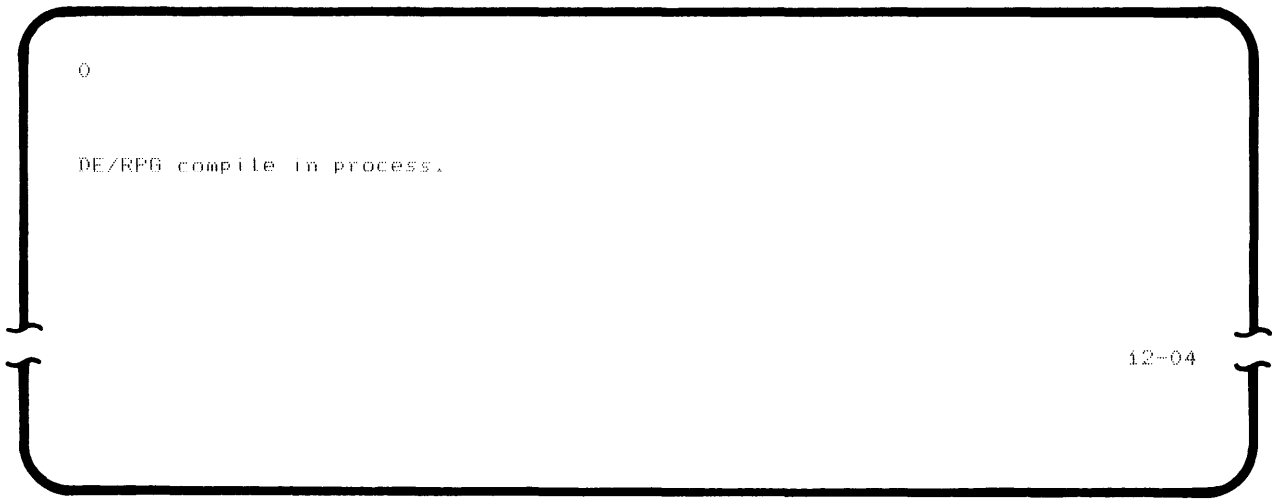
This display provides an opportunity to either reenter the information for the object data set or allocate it. Choose option 2 to allocate the data set. Press the Enter key.

The final display in this example requests you to select the type of listing you want.

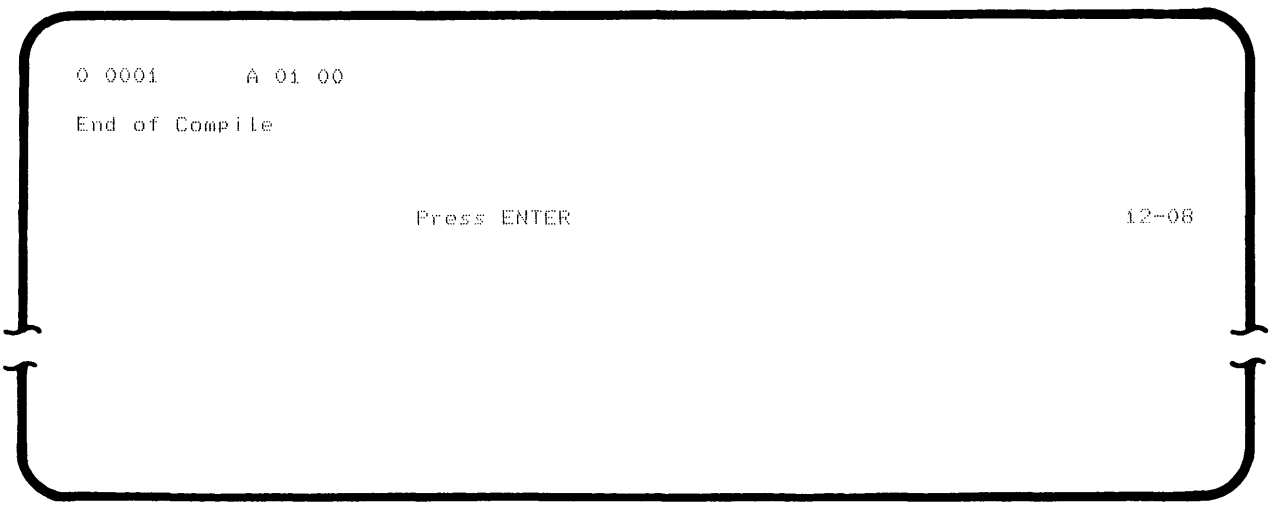
```
0 0001      D 01 40
Select listing option
  1. List to printer
  2. List to diskette
  3. No list
Select option:      Press ENTER                              12-03
```

A printout is provided to show you the appearance of the output when you select option 1 (list to printer). Option 1 is preferred if you have a printer available. If you choose 2, the output is written to a data set on the diskette; however, you must have preallocated two data sets on the diskette to receive the output.

Press the Enter key and the following display appears.



When the compilation has been completed, the following display appears.



Press the Enter key to return to the program load prompt.

You are ready to proceed to Chapter 6 where you will actually use the program you wrote in Chapters 3 and 4 and compiled in Chapter 5.

```
            DE/RPG COMPILER   V0M00

            *Source file.     MASTER
            *Object file.     OBJECT
00001       ZJ  MASTER                                        TFILE(MASTHEAD 10)
00002       Z X1HEADER     E
00003       Z               R                      X1
00004       A           F INPUT        150          DEVICE(CRT) DSPSIZ(6 80)
00005       A           R HEADER
00006       A             CORP          30   I  2  2PMT(ENTER THE NAME OF THE CORPORATI+
00006       A                                        ON) CHECK(DR) DSPATR(UL)
00007       A             CUSN          30   I  2 41PMT(ENTER THE CUSTOMER NAME)
00008       A                                        CHECK(DR) DSPATR(UL)
00009       A             STREET        30   I  3  1PMT(ENTER THE STREET ADDRESS)
00010       A                                        CHECK(DR) DSPATR(UL)
00011       A             CITY          30   I  3 41PMT(ENTER THE CITY)
00012       A                                        CHECK(DR) DSPATR(UL)
00013       A             STATE         20   I  4  1PMT(ENTER THE STATE -- SPELL OUT IN-
00013       A                                         FULL) CHECK(DR) DSPATR(UL)
00014       A                                   O  4 30'CUSTOMER NUMBER:'
00015       A             NUMBER         5   I  4 47CHECK(DR BC) DSPATR(UL)
00016       A                            4   I      CHECK(BY)
00017       A             MARK           1   I  4 80INSERT('H')
00018       A           F MASTHEAD     150          DEVICE(DISK X'4000')

      * ADDR  CONSTANT
      * 02F0  'ENTER THE NAME OF THE CORPORATION'
      * 0311  'ENTER THE CUSTOMER NAME'
      * 0328  'ENTER THE STREET ADDRESS'
      * 0340  'ENTER THE CITY'
      * 034E  'ENTER THE STATE -- SPELL OUT IN FULL'
      * 0372  'CUSTOMER NUMBER:'
      * 0382  'H'
      *
      * ADDR  NAME
      * 0383  CORP
      * 03A1  CUSN
      * 03BF  STREET
      * 03DD  CITY
      * 03FB  STATE
      * 040F  NUMBER
      * 0414  MARK
      *
      *      OBJECT PROGRAM MAP
      *ROUTINE ENTRY POINTS
      *EP      RTN    DESCRIPTION
      *0700    RG99 - End of job processor
      *0764    RG80 - Verify mode error display
      *0824    RG86 - Physical buffer allocation
      *08F0    RG01 - Keyboard external status routine
      *0B24    RG31 - Diskette external status routine
      *0B8C    RG36 - Diskette I/O management routine
      *0C4C    RG50 - I/O driver routine
      *
      *0FE8    Z-spec driver entry point
      *1170    Program entry point
      * 5,632  Is the program length.
```

Chapter 6. Using the Master Customer Identification Job to Enter Data

Make sure the object program is in diskette drive 4000 (or its equivalent for your system). Before you can actually begin entering data using the program you have written, you will have to answer the prompts to the following displays. Use the responses that are shown below.

```
0 0010      N 04 40
Program Name: OBJECT
Device Address: 4000
Partition Number:

                          Press ENTER                          05-00
```

Enter the information that loads the object (OBJECT) program and press the Enter key. The following prompt is displayed.

```
0 0001      D 01 F1
Select initial data entry mode
  Options are
    1. Enter-NEW/ REPLACE    3. Verify      5. Rerun
    2. Update                4. Enter-ADD
Select option:   1  Press ENTER                              06-81
```

Select option 1. You have selected the enter (new) mode on this display because this is the first time you are using the program for this data set.

Press the Enter key and the following display appears:

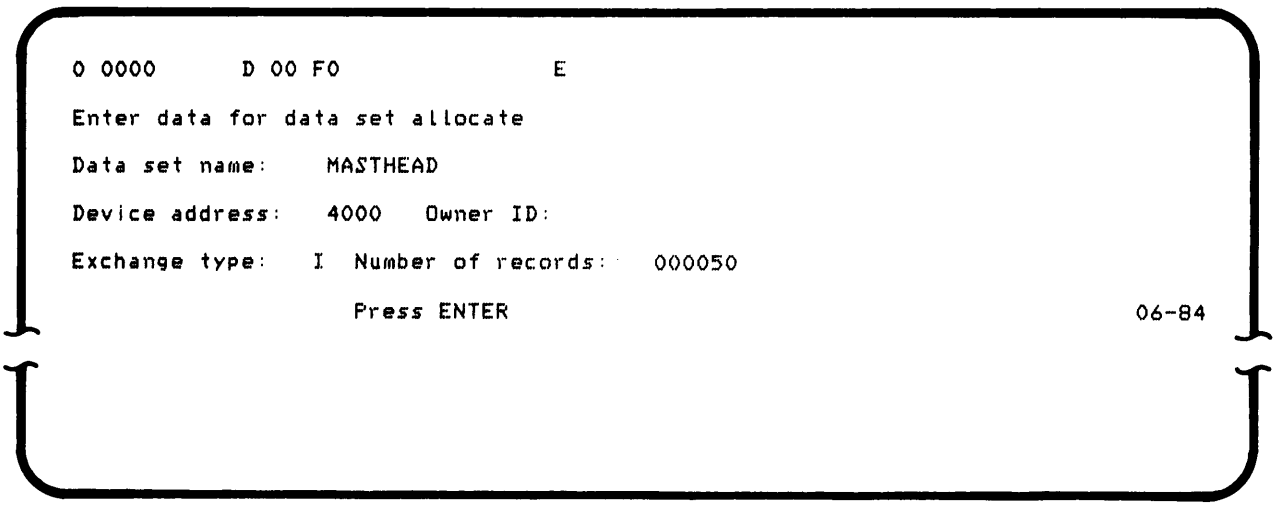
```
0 0001      A 26 D4                E

Enter data for data set open

Data set name:     MASTHEAD

Device address:    4000


                    Press ENTER                                    06-82
```

As you can see, the name you provided for the TFILE parameter in the program is the data set name that is supplied for this display. The address is the same as that for the object program. Accept these values as shown and press the Enter key. The following display appears.

```
0 0001      D 01 40                E

Data set open failed

  Options are

    1. Retry

    2. Allocate

Select option:      Press ENTER                                    06-83
```

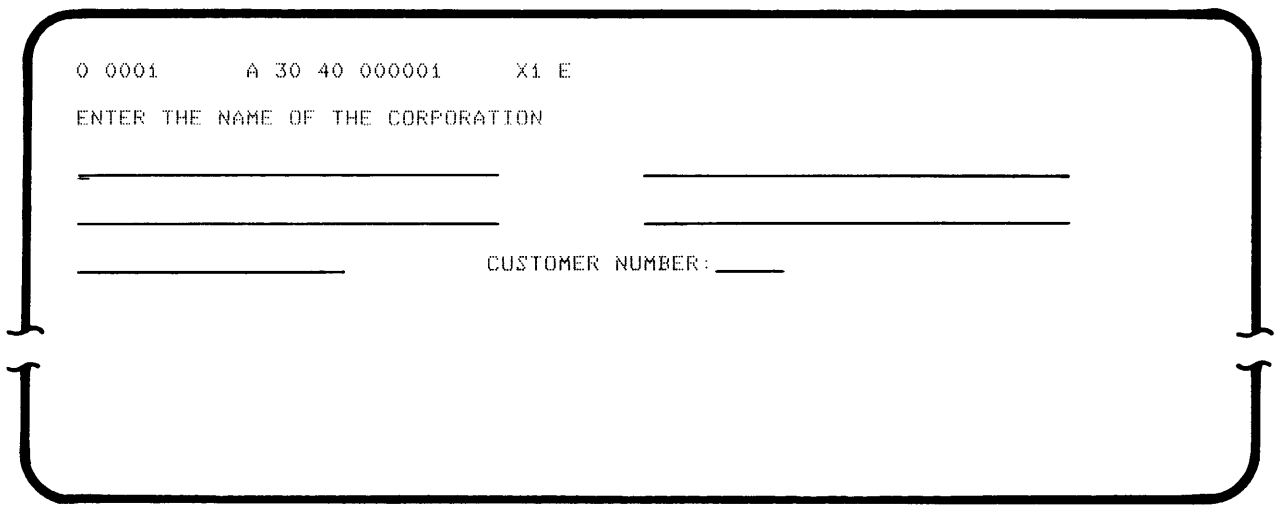
Select option 2 to allocate the MASTHEAD data set.

The following prompt is displayed.

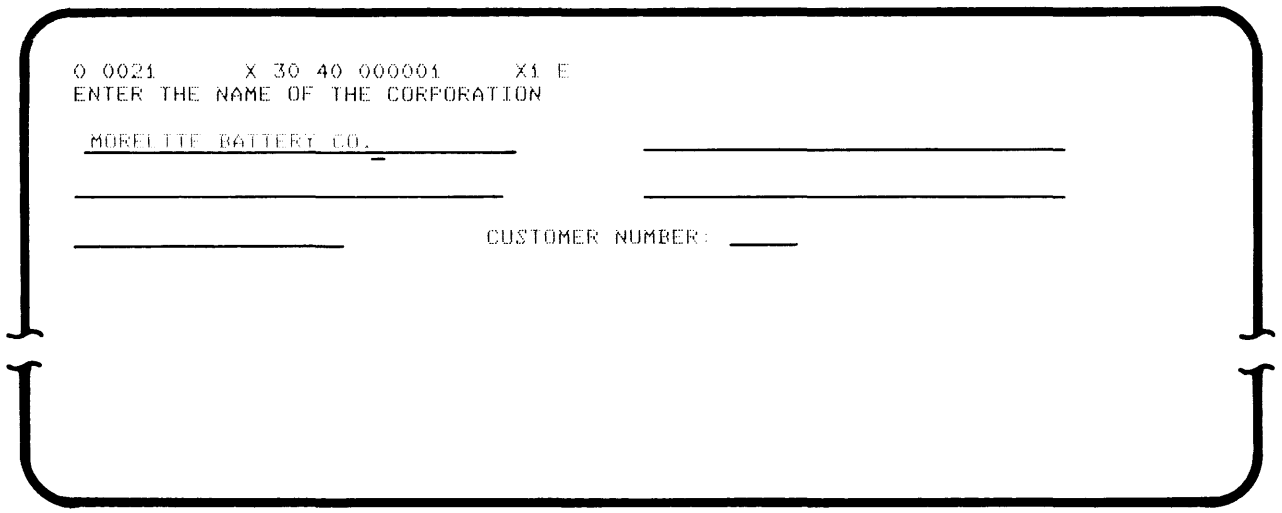


In the number of records field enter 50 and use the field exit key. The number 50 is arbitrary. It will provide 50 records for you to use in entering data using the OBJECT program.

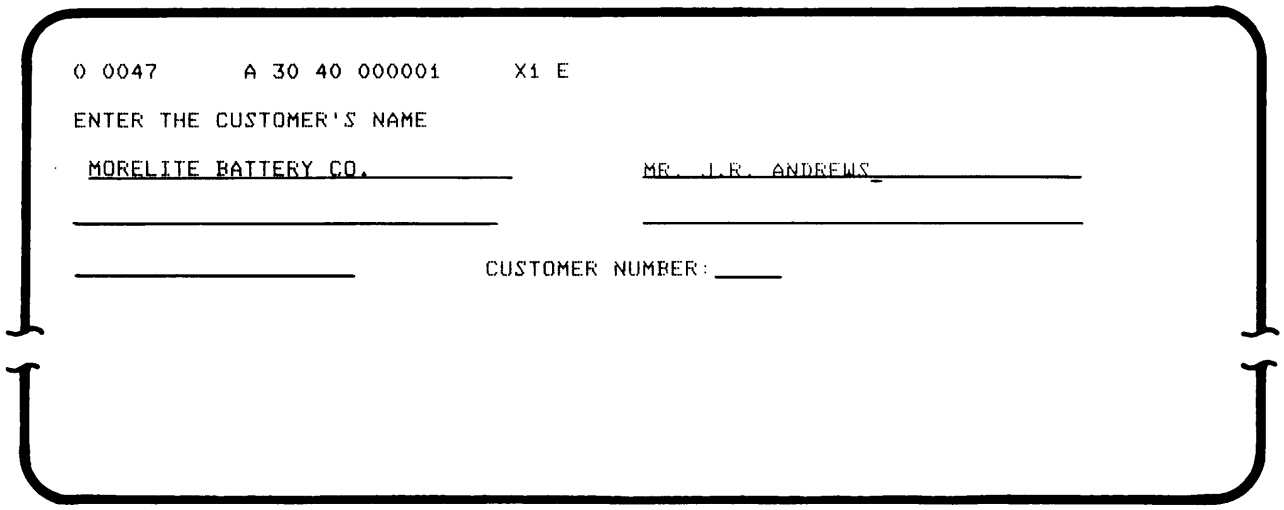
Press the Enter key. The first display of the program appears:

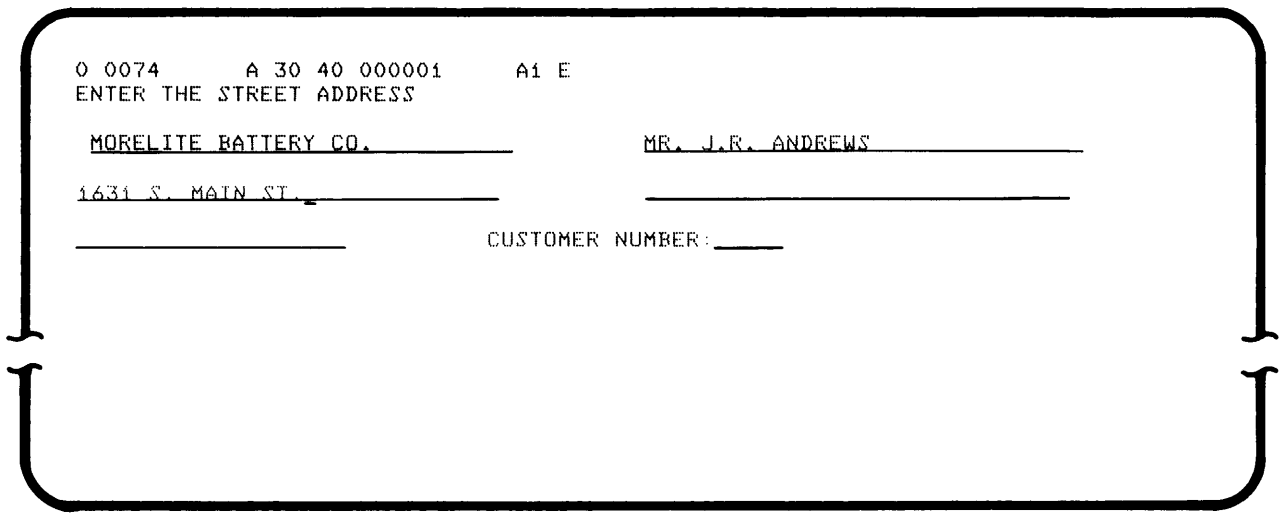


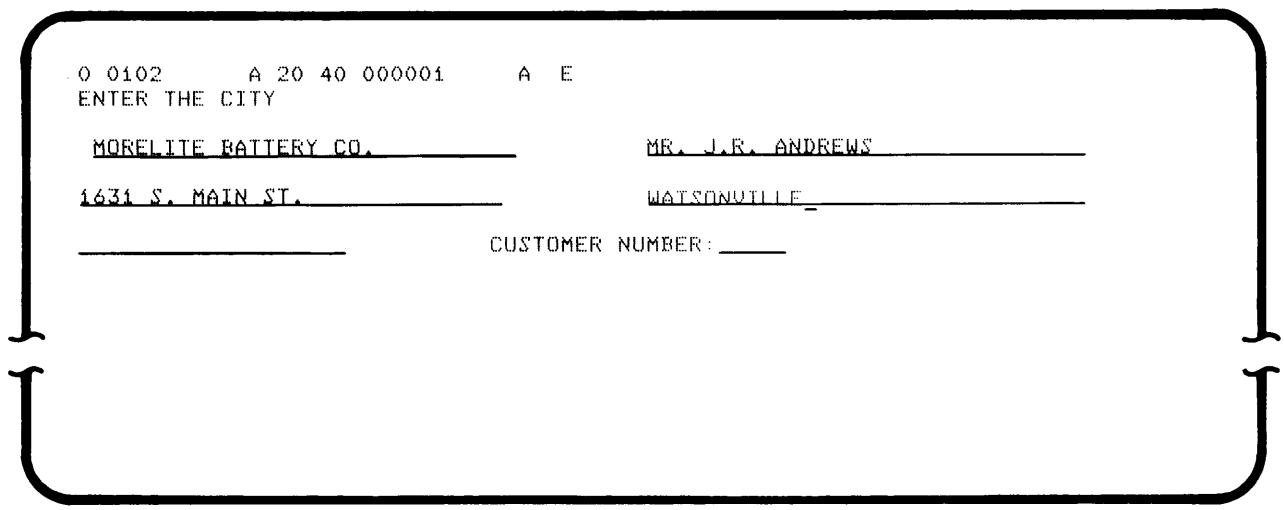
Notice that the prompt is on line 2 of the display and the cursor is in the first position of the first field. Try to exit the first field without entering data. You are not allowed to bypass the field without an entry because you specified the CHECK(DR) edit against the field. An error flashes on the status line. Press the Reset key and return to the first position of the field. Enter the data shown in the following display.

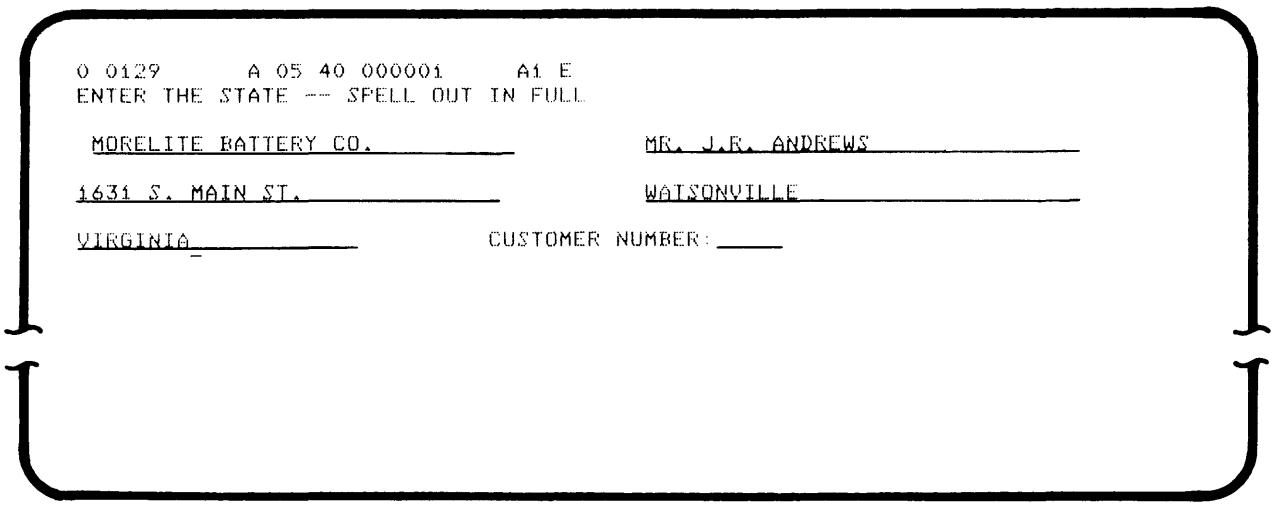


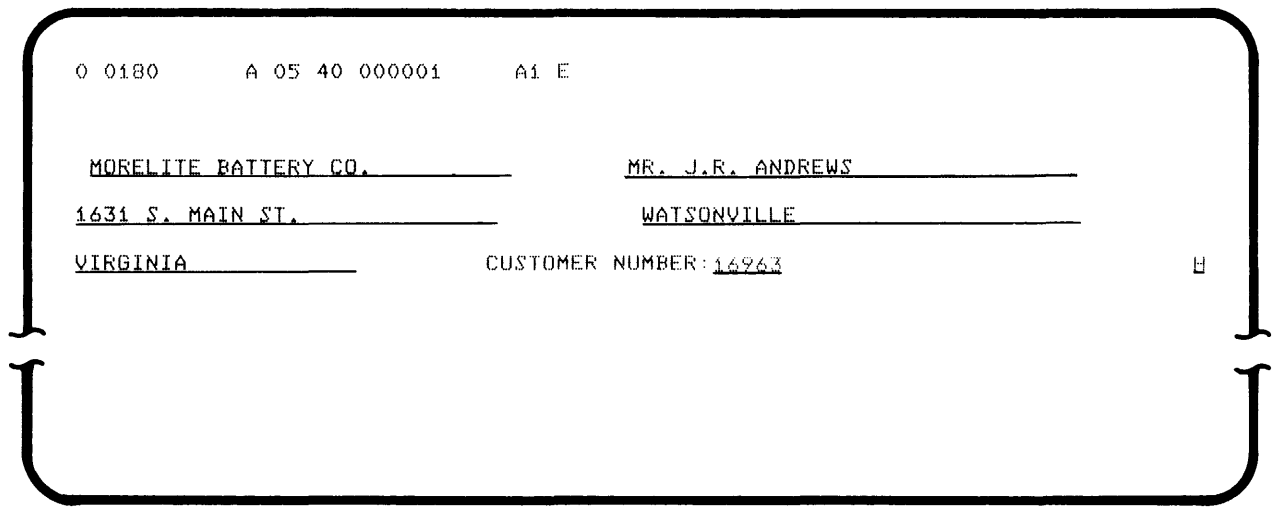
Use a field exit key to advance to the next field. Enter the information on the displays as shown. Use a field advance key after each field.


0 0047 A 30 40 000001 X1 E
ENTER THE CUSTOMER'S NAME
MORELITE BATTERY CO.
MR. J.R. ANDREWS
CUSTOMER NUMBER:


0 0074 A 30 40 000001 A1 E
ENTER THE STREET ADDRESS
MORELITE BATTERY CO.
MR. J.R. ANDREWS
1631 S. MAIN ST.
CUSTOMER NUMBER:


0 0102 A 20 40 000001 A E
ENTER THE CITY
MORELITE BATTERY CO.
MR. J.R. ANDREWS
1631 S. MAIN ST.
WATSONVILLE
CUSTOMER NUMBER:


0 0129 A 05 40 000001 A1 E
ENTER THE STATE -- SPELL OUT IN FULL
MORELITE BATTERY CO.
MR. J.R. ANDREWS
1631 S. MAIN ST.
WATSONVILLE
VIRGINIA
CUSTOMER NUMBER:


0 0180 A 05 40 000001 A1 E
MORELITE BATTERY CO.
MR. J.R. ANDREWS
1631 S. MAIN ST.
WATSONVILLE
VIRGINIA
CUSTOMER NUMBER: 16963

When the last field has been completed, press the Enter key and the next record is automatically displayed. To review records that you have completed, press the Home key. If the cursor is not in position 1 of the record, it will first return there; press the Home key again and the previous record will be displayed. Press the Enter key to advance to the next blank record for entry. If you want to change data in any record, simply enter the changes and press the Enter key.

Once you have completed this process, use the three sample address cards shown in Figure 5 as your source documents. Enter the information from these cards using the program you wrote, entered, and compiled. If you would like to see the contents of the diskette for the MASTHEAD data set, use the Print Utility. The *Utilities Reference/Operation Manual* provides information about using this utility.

When you are done, exit with the EOJ key and try to answer the summary questions.

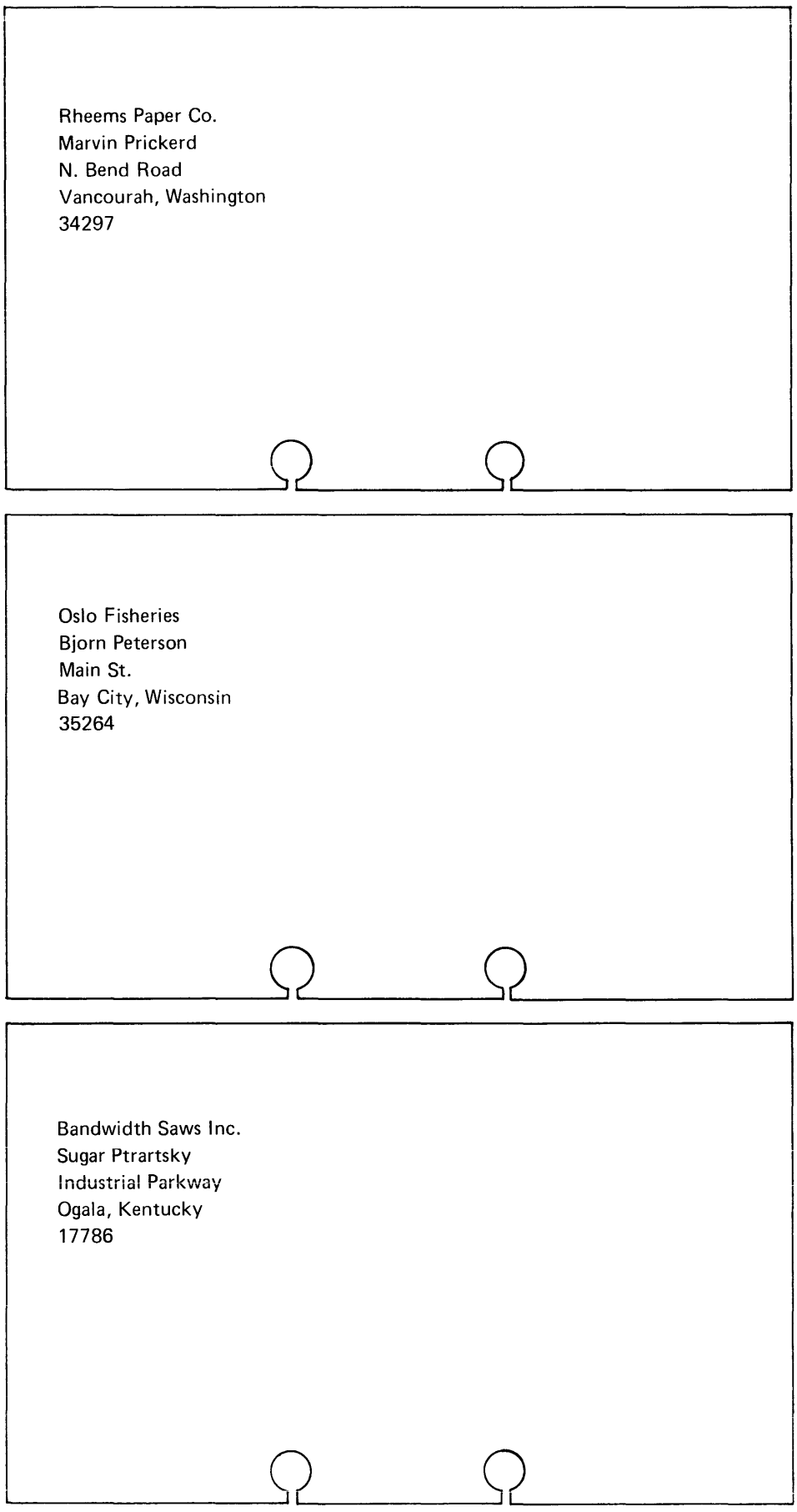


Figure 5. Sample Cards

SUMMARY OF CHAPTERS 2 THROUGH 6

You have created, described, compiled, and used a simple data-entry job. You should understand that the normal description required for a job is:

1. Job specification statement to name the job and the transaction file (Z specification)

2. Entry format statement to name the record to be used for the entry format (Z specification)

3. Review format statement to name the record to be used for the review format (Z specification)

4. File description statement to name and describe the display (CRT) file (A specification)

5. Record description statement to describe the record named in the entry format statement (A specification)

6. Field description statements to describe the characteristics of the fields in the record and to control the keyboard shift and the display organization for each field (A specification)

7. Diskette file description statement to complete the description of the data set for the transaction file (A specification)

The sequence of these statements (as shown in the previous list) is important when you enter and compile your program.

The following chapters describe advanced functions that are available for DE/RPG data-entry jobs. If Chapters 2 through 6 have given you sufficient information to complete your applications, you should be able to answer the following questions and try to code the data-entry program described later in this manual. If you can successfully complete these tasks, you are ready to start writing your own programs.

If you need to learn additional data-entry functions before you can complete your application, read the remaining chapters in this manual. These chapters show you how to display formats but not write them to the diskette data sets, test positions of records to select formats for displaying data for review (update, verify, or rerun modes), use tables, perform automatic calculations, perform additional checks and edits against entered data, and reformat the diskette data set.

Answer the following questions to test your understanding of DE/RPG in simple data-entry jobs:

1. Answer T for true and F for false for each of the following statements. The minimum requirements for a data-entry program are:

   a. _____1 job statement with TFILE specified

   b. _____1 job statement without TFILE specified

   c. _____1 entry format

   d. _____At least 2 entry formats

   e. _____At least one review format

   f. _____1 CRT file

   g. _____Multiple CRT files

   h. _____At least one record

   i. _____More than one record

   j. _____A maximum of 5 fields

   k. _____At least one prompt

   l. _____At least one edit or check

   m. _____1 diskette file

   n. _____Multiple diskette files

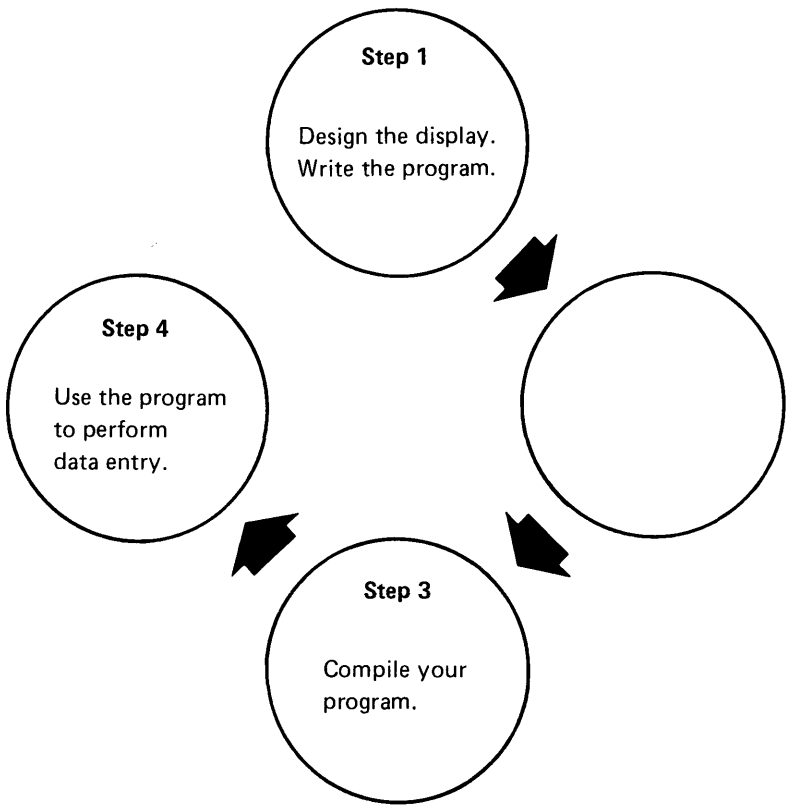
2. Which of the following statements is true?_____

   a. A file is the data that is written on the diskette.

   b. A file is related to an input/output device.

   c. A file is the contents of a display.

3. The following figure duplicates the illustration of the process involved in a simple data-entry job. All circles except step 2 are filled in. Fill in step 2.



4. Which of the following statements is true? _____

   a. An entry format must have the same name as the CRT file.

   b. A format determines the sequence of information on the display.

   c. Formats can be manually selected by the operator or automatically selected by the program.

5. Write the prompt that will display: ENTER THE FOLLOWING INFORMATION.

6. Which of the following statements is true? _____

   a. Record markers illustrate the location of erroneous data.

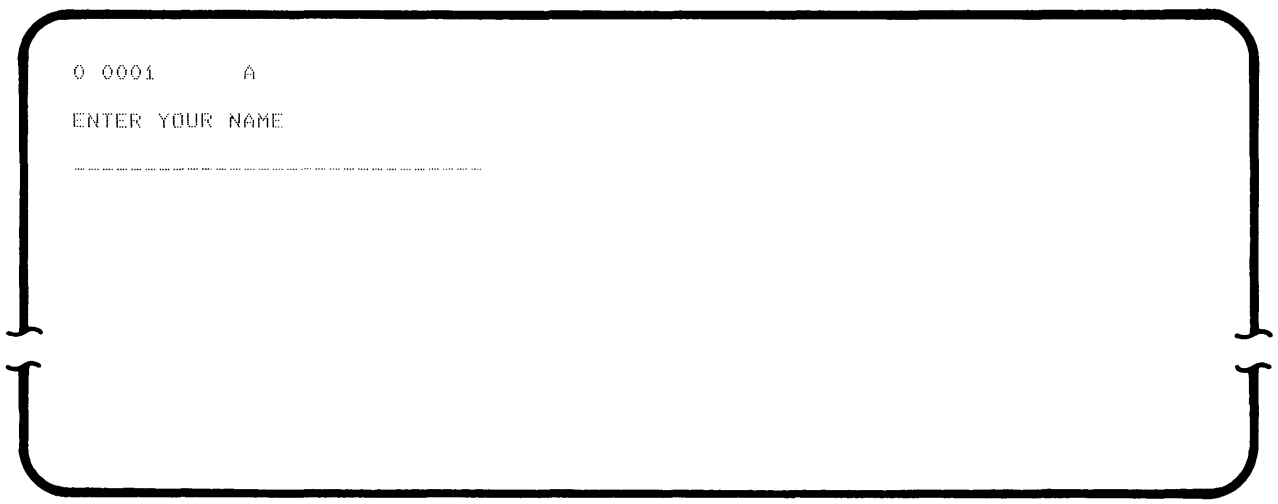
   b. Record markers indicate the type of record.

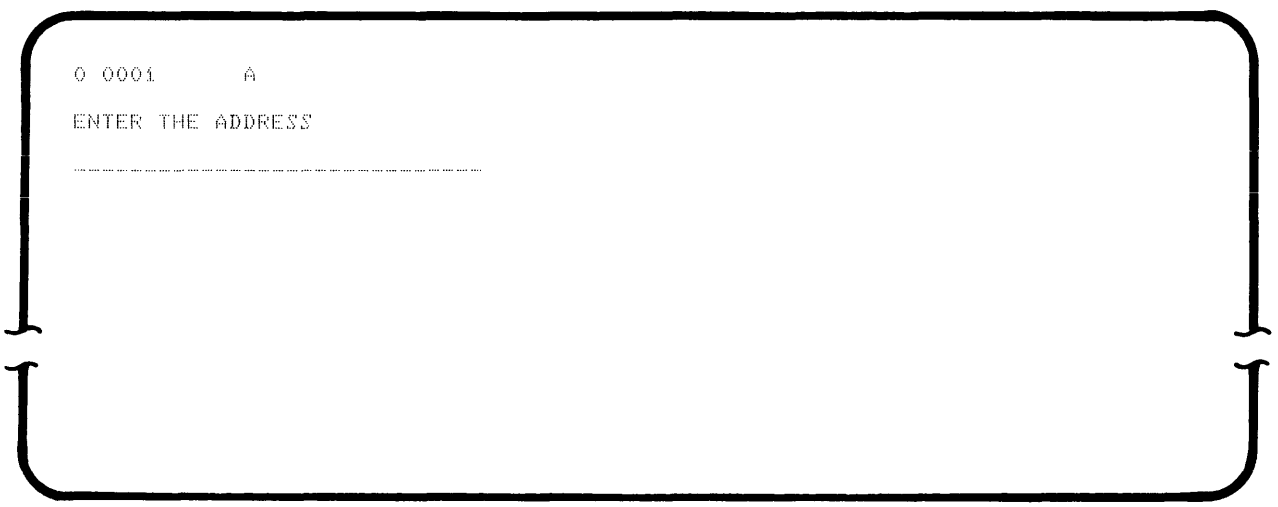
   c. There can only be H and D (for header and detail) record markers in a program.

7. The following program is partially completed. Use the instructions that have been provided to determine how to supply the information needed as shown by the darkened areas on the specifications.

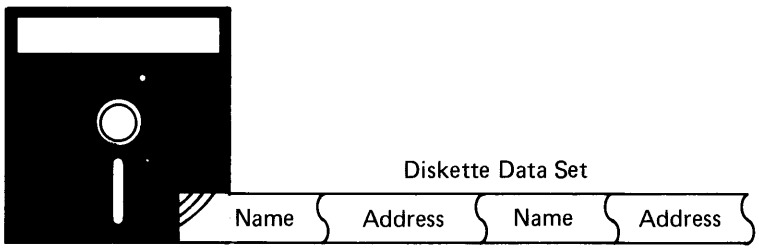
Instructions for the Test Program

This program allows an operator to enter customer name and address records in a diskette data set. When the operator uses the program, the display looks like this:





The resulting data set looks like this:



Using this information, try to fill in the darkened areas of the specifications.

IBM International Business Machines Corporation

IBM 5280 DATA DESCRIPTION SPECIFICATIONS

Printed in U.S.A.

Job No.	Dataset	Keying Instruction	Graphic							
Operator	Date		Key							

Source Document: TEST SAMPLE PROGRAM IN CHAPTER6 | Page of

A

Sequence	Form Type	Comment (*)	Reserved	Indicator (for CHECK (BY, BV) or ERROR)	Reserved	Name Type (F/K/R/T)	Reserved	Dataset/Record/ Field/Table Name	Reserved	Length	Data Type	Reserved	Decimal Positions (0-9)	Usage (I/O/B/W)	Location: Line	Location: Pos	Editing
1	A					F		INPUT		30							DEVICE( ) DSPSIZ(6 80)
2	A					R		NAME									
3	A									30				I			PMT(ENTER YOUR NAME)
4	A																DSPATR( )
5	A																CHECK( )
6	A							ADDRES									
7	A																PMT( )
8	A																DSPATR(UL)
9	A																CHECK(DR)
10	A					F											DEVICE(DISK X'4000')
11	A																
12	A																
13	A																
14	A																
15	A																
	A																
	A																
	A																
	A																
	A																

Checks=CHECK (code . . .)

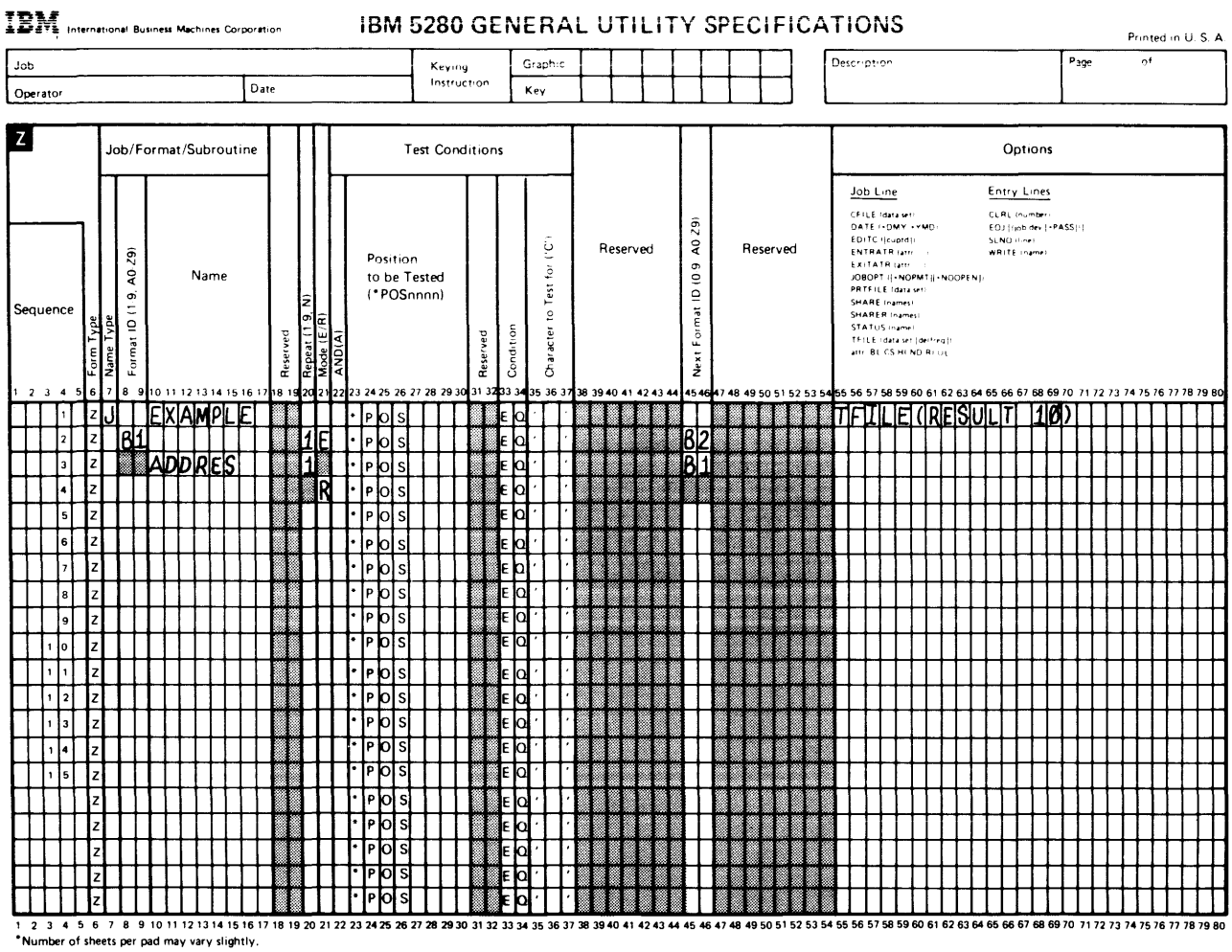
Auto Dup	AD	Mandatory Entry	ME
Auto Skip	AS	Mandatory Fill	MF
Blank Check	BC	Rt Adj-Blank Fill	RB
Bypass	BY	Right to Left	RL
Bypass on Verify	BV	Rt Adj-Zero Fill	RZ
Data Required	DR	Self Check	*nnn
Dup Disable	DD	*C,G (Check Ger	
Field Exit Required	FE	**Modulus	
Lower Case	LC		

Functions

ADD (name)	PMT (prompt)
AUXDUP (name)	RANGE (low high)
AUXST (name)	RANGET (table [index])
COMP (*test fld1'@ fldn [indicator])	RESET (+TOTn)
DSPATR (*attr)	SEQ (*test)
EDTCDE (code [fcat])	SETOF (ind)
ERROR (code [,message])	SETON (ind)
EXSR (subroutine)	SHIFT (*shift)
INSERT (fld1'@ fldn)	SUB (name)
LOOK (table [index,])	SUBST (table1 table2 [index])
	TADD (+TOTn)
*test - EQ,GE,GT,LE,LT,NE	TSUB (+TOTn)
*attr - BL,CA,CS,HI,ND,RI,UL	XCHK (table index1 index2)
'@' - *	*literal'
*shift - A,D,H,N,V,W,X,Y	

1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20 21 22 23 24 25 26 27 28 29 30 31 32 33 34 35 36 37 38 39 40 41 42 43 44 45 46 47 48 49 50 51 52 53 54 55 56 57 58 59 60 61 62 63 64 65 66 67 68 69 70 71 72 73 74 75 76 77 78 79 80

*Number of sheets per pad may vary slightly.



*Number of sheets per pad may vary slightly.

Check your answers with the answers given in Appendix A. If they do not match, try to understand why. Reread the preceding chapters if you need to review the concepts being tested.

Chapter 7. The Description of the Detailed Purchase Job

The initial assignment you were given (in Chapters 2 through 6) was to design a data-entry job that allows an operator to enter information from a customer name and address card file. Your second assignment is to design a job that uses the data set created by the first job. Part of this task will involve defining the current job in such a way that information from the first job can be copied and included as part of the new data set.

Many of the concepts that you learned in the first sample job are also used in this sample job. You will be learning some new automatic functions and you will be learning how to use tables. If you have not understood the information presented in the preceding chapters, reread the necessary portions of the manual. Before you begin this chapter, you should understand:

- The process involved in defining a simple data-entry job
- How to design displays for a data-entry job
- How to write a simple program using the A and Z specifications
- How to enter, compile, and use the program you wrote

If you understand these concepts, continue reading about the advanced data-entry functions offered by DE/RPG.

DEFINITION OF THE DETAILED PURCHASE JOB

The environment for this sample job is the same as that used for the first job. Assume that you are part of the same data-entry department that created the first job. The manager of the sales department has requested that your department design a job for entering data from a sales order form that his salespeople use. The sales order form is shown in Figure 6. Read the instructions carefully. Notice that the customer information at the top of the sales order form is identical to the type of information you entered for the first job.

3340 Silver Bark Drive
Oak Ridge, Montana

ACKME PARTS COMPANY

Parts Suppliers with a Name of Excellence

Date _____

Customer's Name _____

Salesman's Initials _____

Address _____

ITEM NUMBER	DESCRIPTION	QUANTITY	UNIT PRICE	

Customer Number: _____

Figure 6. Sales Order Form

Your supervisor has given you a sheet of instructions describing the job.

Jobname: TODAY DATASET NAME: BILLING — Data Set Created by this Job
Master Customer DATA SET Name: MASTHEAD — Data Set Used by this Job

Record Type 1. HEADER Record:
Copy entire Header record from master. Use the customer # to find the correct Header.

2. DETAIL Record: (Enter date from Sales form)

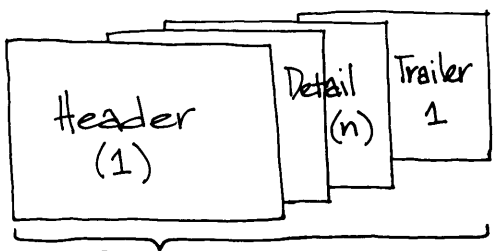
Field Name CUSTNU(5) – Customer number. Same as on header.

Field Length SALS(3) – Salesman's Initials. Alphabetic only
ITEM(6) – Item #. Pattern DDXXXD. Check against the ITEMT table. Use index A. Cross-check against the INVENT table.
DESC(30) – Description. Free-form.
QUANT(4) – Quantity. Digits, 0 decimal positions, right adjust with zero fill.
PRICE(5) – Digits, 2 decimal positions, right adjust with zero fill, Check against the PRICET table. Use index B. Cross-check against the INVENT table.

Editing Instructions COST(9) – Unit Price. Digits, 2 decimal positions, MULTIPLY PRICE * QUANT to obtain. Keep online tot in *TOT1.
MARK2(1) – D mark in position 150 of diskette record.

3. TRAILER Record: (Automatically generate)
DAY(6) – Date – current date
CUSTNU(5) – Customer number. Same as Header and Details
TOT(15) – 2 decimal positions (contains online total for all details).
MARK3(1) T mark in position 150 of the diskette record

View of the Resulting Data Set



For each customer who made a purchase

Read these instructions carefully. Notice that you are told to copy customer information from the data set created by the first sample job and then enter the purchase data from the sales order forms that have a matching customer number. Each order line on the sales order form equals one detail record. These records will be marked with a D in the last position of the diskette record to indicate that they are detail type records.

When all detail records for a customer have been entered, the system automatically computes the total and puts this total in a separate record marked with a T to indicate that it is a trailer record.

Because each record is to be 150 positions long, the record marker for each record type will be in position 150. By requiring that all record types have record markers in the same position, you can ensure that the correct format will be selected for the record.

Suppose for instance, that the header record contained an H in position 150, the detail record contained a D in position 69, and that the trailer record contained a T in position 30. Now, suppose the program selected the format for displaying detail records whenever there was a D in position 69 of the diskette record. If the header record contained a D in that position, a header record would be displayed with a detail record format. By placing the marker for all record types in the same position of the diskette data set, you can prevent this type of error.

The use of header, detail, and trailer record types is a common technique in data-entry applications. Figure 7 shows the kind of information contained in each record type.

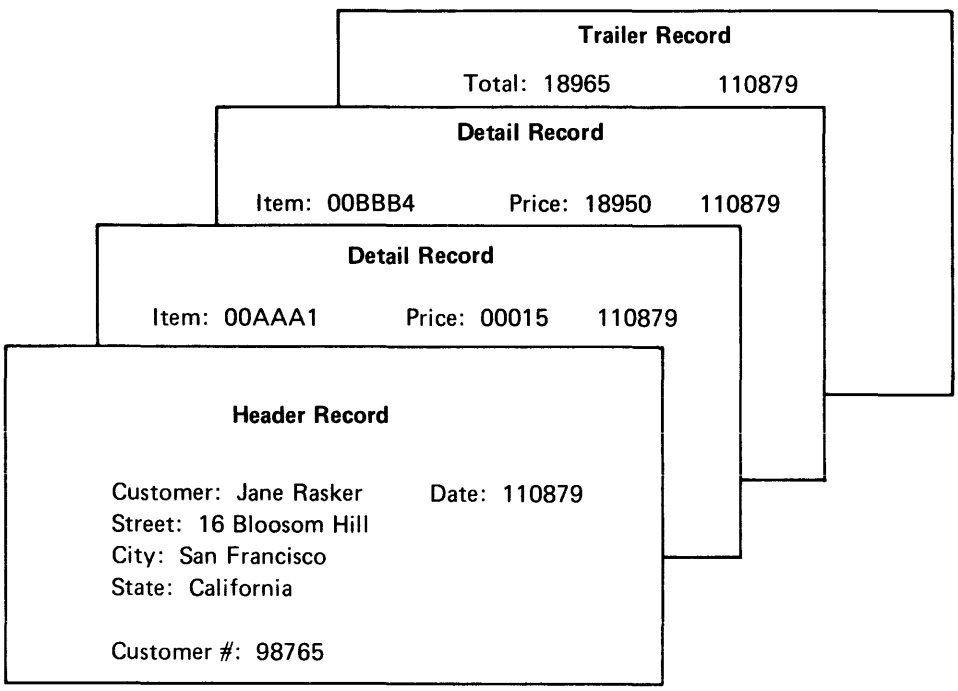


Figure 7. Record Types

DEFINING THE REQUIREMENTS FOR THE DETAILED PURCHASE JOB

The job you are now preparing to define is the one that allows the operator to enter detailed purchase information from the sales order form. The fields in the two new records are:

Display fields on The Sales Order Form	**Diskette fields In the Instructions**
Detail Record	
Date	Date
Customer number	Customer number
Salesman's Initials	Salesman's Initials
Item number	Item number
Description	Description
Quantity	Quantity
Price	Price
	Cost
	D
Trailer Record	
	Date
	Customer number
	Total
	T

The display and diskette formats do not match. Therefore, the diskette format will be the guide for the field design because it contains additional fields that must be included on the displays that are not shown on the sales order form. The new functions used in this job are the use of tables and some job, format, file, and field functions not described in the first sample job.

The new concepts that you will find in this sample job are:

- Using multiple record types and multiple data sets
- Automatically calculating totals
- Automatically duplicating fields
- Using tables
- Suppressing the writing of a record
- Reformatting a record for the diskette

BASIC CONCEPTS FOR THE DETAILED PURCHASE JOB

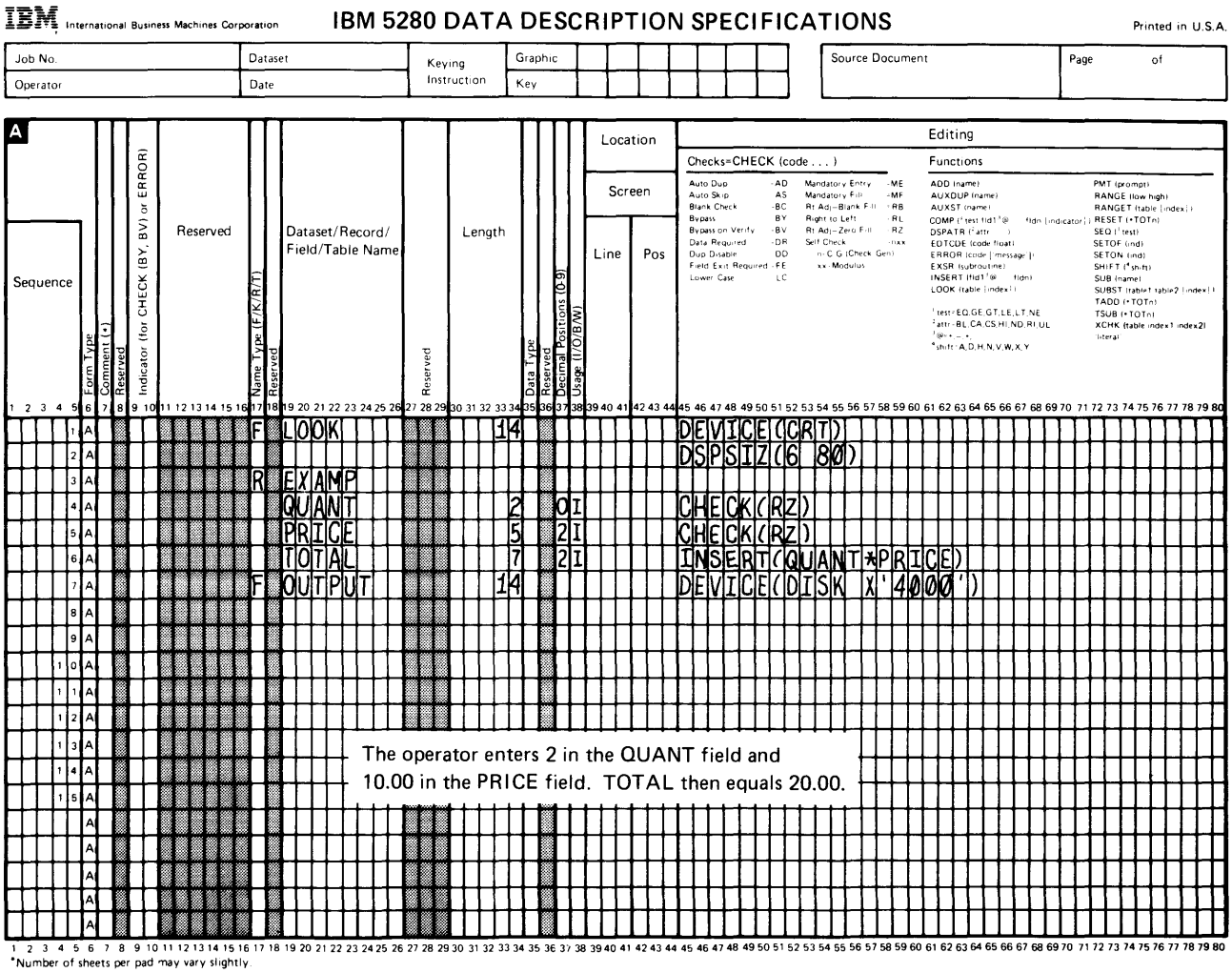
Two concepts that you need to understand before you begin describing the fields in this job are (1) the use of named fields and (2) the use of tables and indexes.

Named Fields

A named field is any field that has a name in columns 19 through 24 of the A specification. You can also specify a named field by using the AUXST keyword and naming the field in the parameter (for example, AUXST(DATE)), instead of columns 19 through 24. Remember that the supervisor instructed you to name each field in the first example; therefore, each data field in that program was a named field. Naming fields allows you to refer to them later in the same program. If a field is not named, the entry can only be used in the record in which it was entered. Therefore, it is a good technique to name any fields in your program you think you will need to use any place other than their initial entry.

Basically, you can use named fields (1) to duplicate data from one field into another (2) to perform calculations, or (3) to insert the contents of a field into another.

When you name a field, DE/RPG places the data associated with that field in storage. It can retrieve the field any time you use the name with a keyword or reserved word in columns 45 through 80 of the A specification. For example, if you name one field PRICE and another field QUANT, you can specify INSERT(PRICE*QUANT) in a third field (TOTAL) and the result of multiplying the contents of PRICE and QUANT is placed in the TOTAL field.



A reserved word is a name that is restricted for special uses such as using counters (*TOT) or specifying test positions (*POS). You will learn more about the reserved words *TOT and *POS in the second sample program in this manual. The *DE/RPG Reference Manual* contains descriptions for all reserved words.

Tables and Indexes

Tables are lists of data that are used to determine the accuracy of entries. The types of checks that can be performed against tables are:

- Checking for matching data
- Checking for an acceptable range of data
- Checking entries for matching combinations of data
- Substituting the contents of the entered field with an entry from a table

An example of a table is a list containing identification codes for items sold by a business.

ITEMT Sample Table	Item Description
00AAA1	Ratchet assembly
00AAA2	1-inch screws
00AAA3	Roller bearings
00AAA4	3-hp motors
00AAA5	Spring clamps
00AAA6	Motor housing

The sample table is named ITEMT. A table name can consist of any nonblank character string except one that begins with TAB. For example, if this table were named TABITEM, it would not be accepted by DE/RPG. (This is to avoid conflict with RPG rules.)

The following sample field description statement illustrates the coding for a table function that checks the validity of data entered in the field against entries in a table.

The table function is named LOOK. When the operator enters data into the ITEM field, DE/RPG uses the data to check entries in the table named ITEMT. If it finds a match for the data, the next field is processed. If it does not find a match, an error occurs.

IBM International Business Machines Corporation

IBM 5280 DATA DESCRIPTION SPECIFICATIONS

Printed in U.S.A.

Job No.	Dataset	Keying Instruction	Graphic								Source Document	Page of
Operator	Date		Key									

A

Sequence	Form Type	Comment (*)	Reserved	Indicator (for CHECK (BY, BV) or ERROR)	Reserved	Name Type (F/K/R/T)	Reserved	Dataset/Record/ Field/Table Name	Reserved	Length	Data Type	Reserved	Decimal Positions (0-9)	Usage (I/O/B/W)	Location: Screen Line	Location: Screen Pos	Editing
1	A					F		IN		6							DEVICE(CRT)
2	A																DSPSIZ(6 80)
3	A					R		REC									
4	A							ITEM		6				I			LOOK(ITEMT)
5	A					F		OUT		6							DEVICE(DISK X'4000')
6	A																
7	A																
8	A																
9	A																
10	A																

Checks=CHECK (code . . .)

Auto Dup	AD	Mandatory Entry	ME
Auto Skip	AS	Mandatory Fill	MF
Blank Check	BC	Rt Adj–Blank Fill	RB
Bypass	BY	Right to Left	RL
Bypass on Ver fy	BV	Rt Adj–Zero Fill	RZ
Data Required	DR	Self Check	nnn
Dup Disable	DD	n–C G (Check Gen)	
Field Exit Required	FE	xx–Modulus	
Lower Case	LC		

Functions

ADD (name)	PMT (prompt)
AUXDUP (name)	RANGE (low high)
AUXST (name)	RANGET (table [index])
COMP ([1]test fld1 [2]@ fldn [indicator])	RESET (+TOTn)
DSPATR ([3]attr)	SEQ ([1]test)
EDTCDE (code [float])	SETOF (ind)
ERROR (code [message])	SETON (ind)
EXSR (subroutine)	SHIFT ([4]shift)
INSERT (fld1 [2]@ fldn)	SUB (name)
LOOK (table [index])	SUBST (table1 table2 [index])
	TADD (+TOTn)
[1]test: EQ,GE,GT,LE,LT,NE	TSUB (+TOTn)
[2]attr: BL,CA,CS,HI,ND,RI,UL	XCHK (table index1 index2)
[3]@ + – * /	'literal'
[4]shift: A,D,H,N,V,W,X,Y	

Because this table function did not use an index, the program does not know the number of the table entry that matched the entry for the valid field.

indexes are named numeric fields that contain the number (position) of the entry in the table that matches the field being tested. The following modified sample illustrates the use of an index.

IBM International Business Machines Corporation

IBM 5280 DATA DESCRIPTION SPECIFICATIONS

Printed in U.S.A.

Job No	Dataset	Keying Instruction	Graphic								Source Document	Page of
Operator	Date		Key								TABLE USE SAMPLE	

A

Sequence	Form Type	Comment (*)	Reserved	Indicator (for CHECK (BY, BV) or ERROR)	Reserved	Name Type (F/K/R/T)	Reserved	Dataset/Record/ Field/Table Name	Reserved	Length	Data Type	Decimal Positions (0-9)	Reserved	Usage (I/O/B/W)	Location: Line	Location: Pos	Editing: Checks: CHECK (code . . .)	Editing: Functions
1	A					F		IN		6							DEVICE(CRT)	
2	A																DSPSIZ(6 80)	
3	A					R		REC										
4	A							A		1		0		W				
5	A							ITEM		6				I			LOOK(ITEMT A)	
6	A					F		OUT		6							DEVICE(DISK X'4000')	
7	A																	
8	A																	
9	A																	
10	A																	
11	A																	
12	A																	
13	A																	
14	A																	
15	A																	
	A																	
	A																	
	A																	
	A																	
	A																	

Suppose the ITEMT table consists of the entries shown in the sample table. When the operator enters 00AAA3 into the ITEM field, index A will contain 3 (for position 3 of the table which contains the matching entry).

*Number of sheets per pad may vary slightly.

Notice that the field naming the index precedes the field that uses the index. If indexes are not defined in the program that uses them, they default to numeric fields that are 5 positions long. Things to consider in defining an index are (1) they must be named, (2) they must be numeric, and (3) their field length is determined by the number of entries in the table.

Not all table functions require indexes. The LOOK and XCHK keywords (which are used in the second sample program) do use indexes. The LOOK keyword, as you have seen, compares a single field entry against entries in a table. The XCHK (cross-check) keyword compares the combination of two fields with entries in a table (the entries represent the valid combinations). To perform a cross-check, you must perform two LOOK operations and keep the values of the matching positions in the tables in two different indexes. Then, you use these indexes with an entry in the table that you are cross-checking to determine whether the combination of values is acceptable. The following shows an example of using the LOOK and XCHK keywords. Additional details about this operation are provided when you write the program.

Indexes should have as many positions as is necessary to hold the longest position in the table. For example, the ITEMT table in the sample has 6 entries; therefore, it needs an index with a length of 1 (the index holds position 1, 2, 3, and so on up to 6) and a table with 12 entries needs an index with a length of 2 (this index holds 1, 2, and so on up to 12).

To understand what is happening in this small program, you will need to look at the PRICET and INVENT tables. The ITEMT table has already been described earlier in this topic.

PRICET	INVENT	
08995	1 1	} ITEM 1 sold at price 1
00013	2 2	
00596	3 3	
04379	4 4	
00127	5 5	
00835	6 6	

The PRICET table contains the cost of the item. For example 00596 means the cost is $ 5.96. The INVENT table contains valid combinations of items and prices. For example, the INVENT table specifies that the combination of selling item 1 (00AAA1) at price 1 (08995) is valid.

IBM International Business Machines Corporation

IBM 5280 DATA DESCRIPTION SPECIFICATIONS

Printed in U.S.A.

Job No	Dataset	Keying Instruction	Graphic								Source Document: EXAMPLE OF CROSS-CHECKING TABLES	Page of
Operator	Date		Key									

A

Sequence	Form Type	Comment (*)	Reserved	Indicator (for CHECK (BY, BV) or ERROR)	Reserved	Name Type (F/K/R/T)	Reserved	Dataset/Record/ Field/Table Name	Reserved	Length	Data Type	Reserved	Decimal Positions (0-9)	Usage (I/O/B/W)	Location: Screen Line	Location: Screen Pos	Editing: Checks=CHECK (code ...)	Editing: Functions
1	A					F		IN		11								DEVICE(CRT)
2	A																	DSPSIZ(6 80)
3	A					R		REC										
4	A							A			1			O	W			
5	A							B			1			O	W			
6	A							ITEM			6			I				LOOK(ITEMT A)
7	A							PRICE			5		2	I				LOOK(PRICET B)
8	A																	XCHK(INVENT A B)
9	A					F		OUT		11								DEVICE(DISK X'4000')
10	A																	
11	A																	
12	A																	
13	A																	
14	A																	
15	A																	
	A																	
	A																	
	A																	
	A																	
	A																	

*Number of sheets per pad may vary slightly.

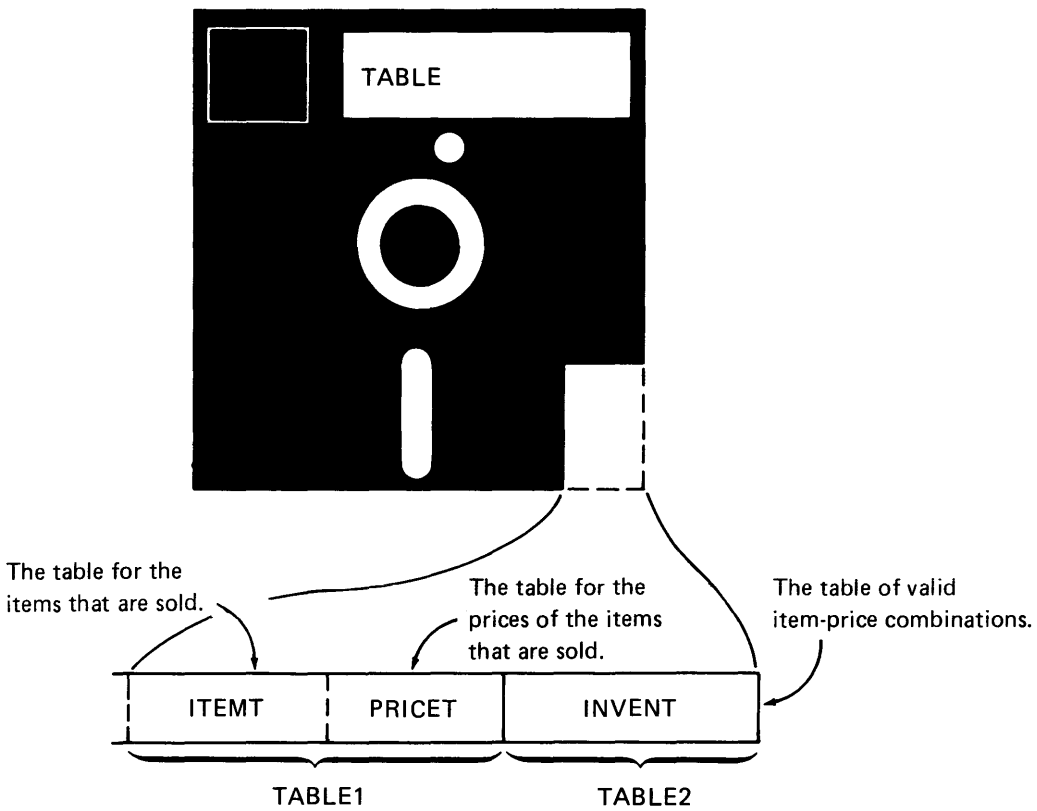
Suppose the ITEMT table consists of the entries shown in the sample table. When the operator enters 00AAA3 into the ITEM field, index A contains 3 (for position 3 of the ITEMT table which contains the matching entry). The operator then enters 00596 in the PRICE field (this matches position 3 of the PRICET table); index B then contains a 3. The cross-check uses the 3 in index A and the 3 in index B to see if the combination of 3 3 is valid for the INVENT table. There is a pair of entries for the combination 3 3 in the INVENT table, so the entries are accepted.

USING TABLES IN THE DETAILED PURCHASE JOB

The instructions tell you to use tables for two purposes in this job:

- To verify that information, such as the item and price, is valid
- To verify that combinations of information (such as the combination of an item and price) are valid

The detailed purchase job uses three tables which are in two table data sets:



Assume that these tables are already available on a diskette. It is your job to simply use the existing tables to verify that information is correct. The next chapter goes through the process of writing the program for the detailed purchase job. Chapter 9 shows you how to create the table data sets used by the detailed purchase job.

SUMMARY OF CHAPTER 7

You have completed the process of reviewing the requirements of the detailed purchase job and of learning about named fields, tables, and indexes which the program will use. Before you proceed to the next chapter, where you will design the displays and write the program, try to answer the following questions.

1. Which of the following statements is true? _____

   a. The detailed purchase job contains three record types.

   b. The detailed purchase job has display formats that match the diskette formats.

   c. The detailed purchase job uses tables.

   d. The source document for the detailed purchase job is a sales order form.

2. Which of the following statements is true? _____

   a. A named field can be used in more than one place in a program.

   b. A table is a special type of named field.

   c. A grocery list can be considered a table.

Check your answers with the answers given in Appendix A. If they do not match, try to understand why. Reread the chapter if you do not understand the concepts being tested.

Chapter 8. Designing the Displays and Writing the Program for the Detailed Purchase Job

You are almost ready to begin marking the display work sheets. Before you do, you should be aware of two new techniques in this program that affect the display records.

The first technique is the use of a scratch record. This is a special kind of record that is only displayed and never written in the diskette data set. It will be the first record displayed in the program. The scratch record will allow the operator to review the instructions for using the program and to set up the date fields for the detail records.

The scratch record provides operating instructions that guide the operator in using the program. It also provides an input field that the operator can use in other records. The input field is for the date on the sales order form. If there are many orders on a form, the operator can enter the date once in the scratch record and then automatically duplicate it in the detail records. Later, you will learn how to describe the date fields in the scratch record and in the detail record so they can be changed or duplicated as needed.

The second technique consists of reformatting diskette records by describing them after the file description statement for the diskette. In the first sample program, you used the CHECK(BY) operation to accomplish the same thing that this reformatting will accomplish—the placement of the record marker. Reformatting records, as used in this sample, affects the names you can give the display records. Each record name in a program must be unique. The names for the diskette records are specified in the instructions: DETAIL and TRAILER. You must invent new names to use for the display records. The sample arbitrarily uses DET for the detail display record and TRAIL for the trailer display record.

DESIGNING THE DISPLAYS

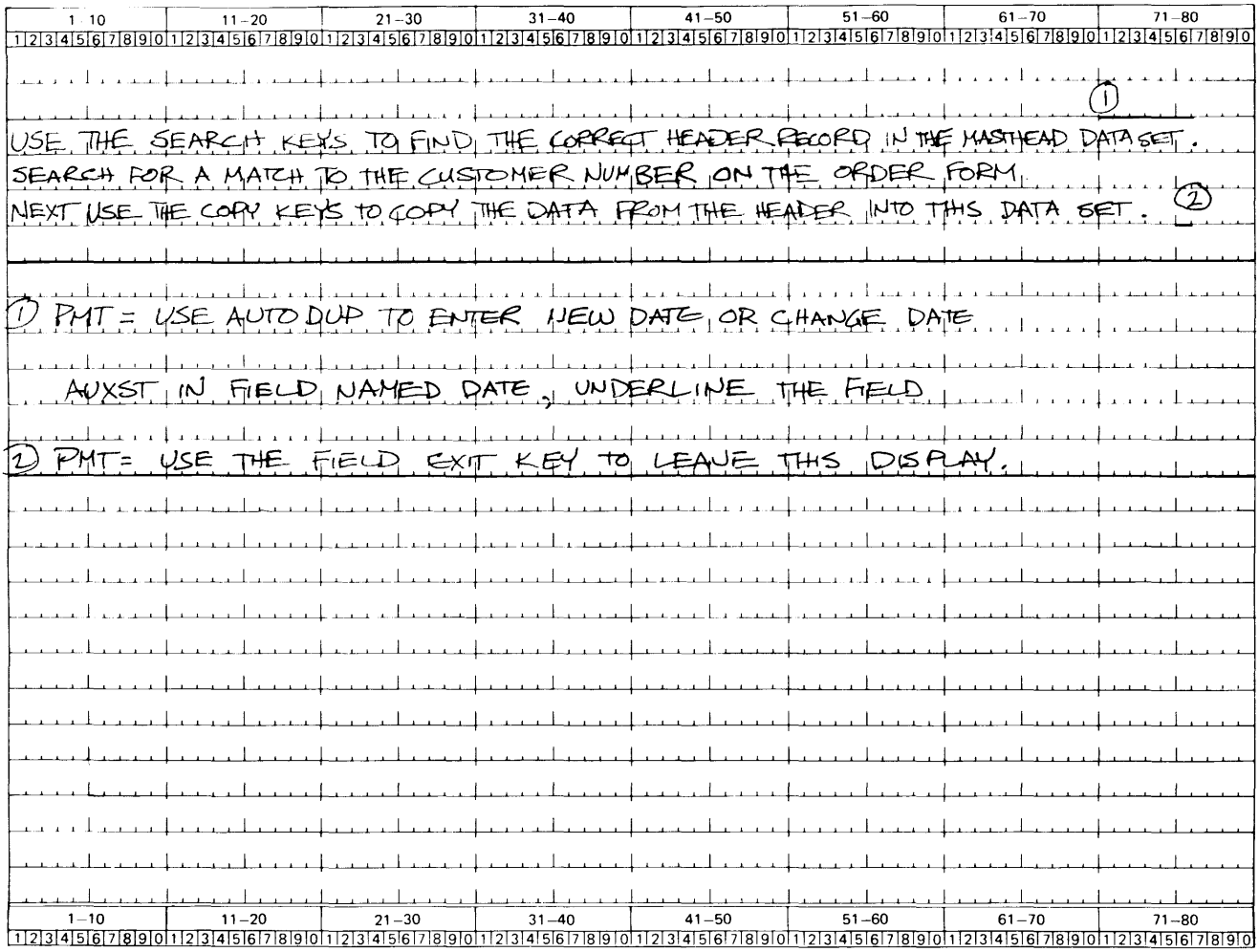
As you did in the first sample job, begin this job definition by designing the displays that will be used. Be sure to mark the display work sheets with the characteristics of the fields. When you have completed the design, read the following text to see whether your design and the sample design match. If your design and the sample do not match generally, try to determine where they differ and why. You may have added checks and edits that the sample does not use. This is acceptable as long as you describe on the A specification all checks and edits that the sample uses. If you have difficulty with this step, reread Chapter 3.

The notes on the display work sheets reflect the instructions provided by the supervisor. The additional functions that are included in the sample depend on your experience in data-entry tasks.

Look at the sample work sheet that follows.

Display Screen Layout Sheet

COLUMN



Literals are used on the sample work sheet to provide instructions for the operation of the program. The advantage of literals over prompts is that the literals allow you to use the entire display. The operator can carefully read the instructions until he uses the Record Advance key. Pay close attention to the length allowed by the 80 column displays. Do not divide words; they might be difficult to read and understand. One way to avoid dividing words is to keep each literal field to a maximum of 80 characters long.

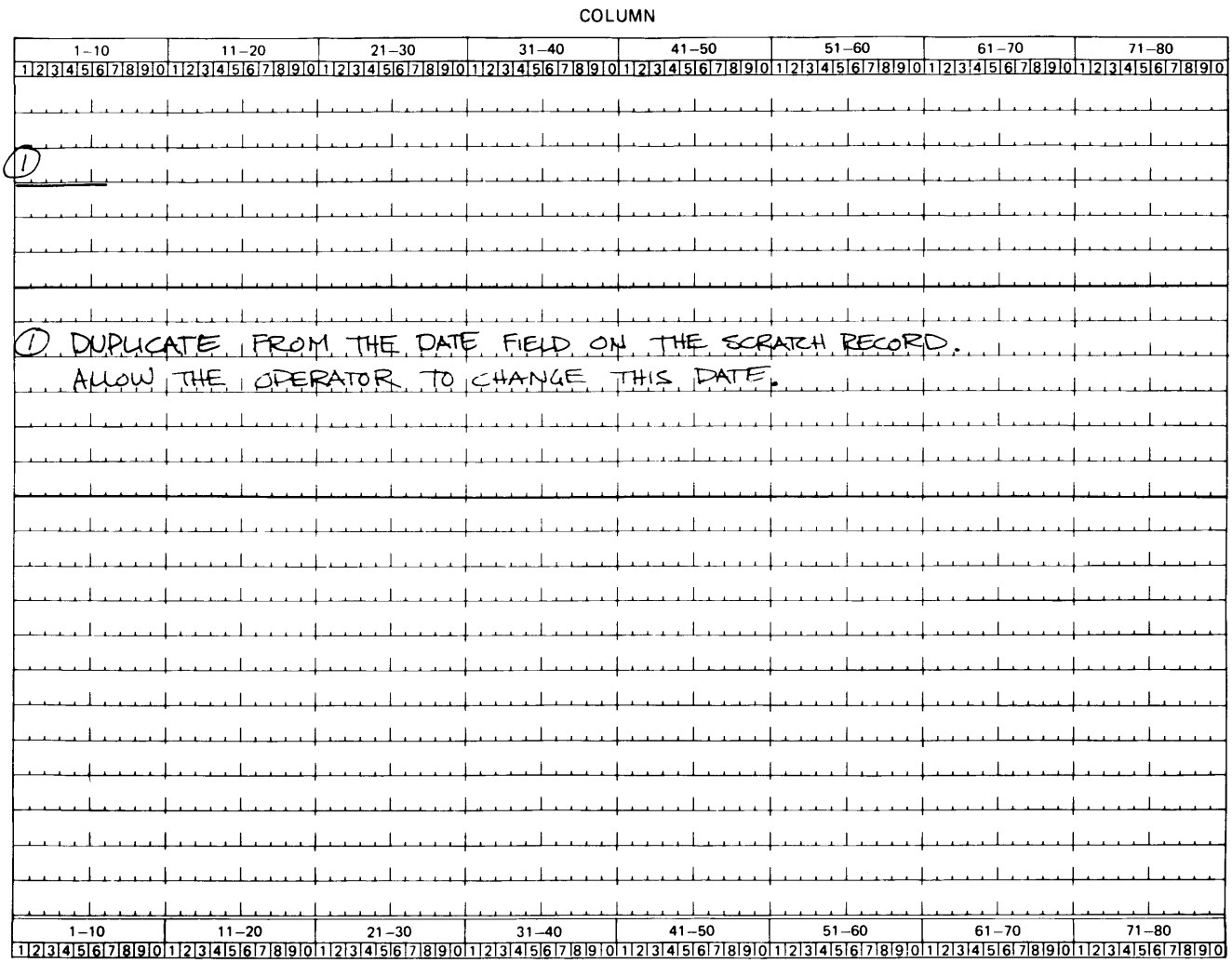
The sample work sheet shows that a single-position field has been added to the scratch record. This field has a prompt that instructs the operator to use the Field Exit key to leave the display. If the operator were using the automatic record advance function and did not include this field, the display would flash past the operator without halting. The operator would not see the instructions and would not know what to do. With this field, the operator controls the display; when the operator uses the Field Exit key as prompted, the display disappears and another display replaces it. You will learn how to specify this field when you write the program.

The scratch record has been described. The scratch record contains only a date field, prompts to guide the operator, and a single-position field to allow the operator to end the display. When you use the A and Z specifications to write the program, you will learn how you can display this scratch record without writing it in the diskette data set.

Now you are ready to look at the display work sheets for the detail display (DET) record.

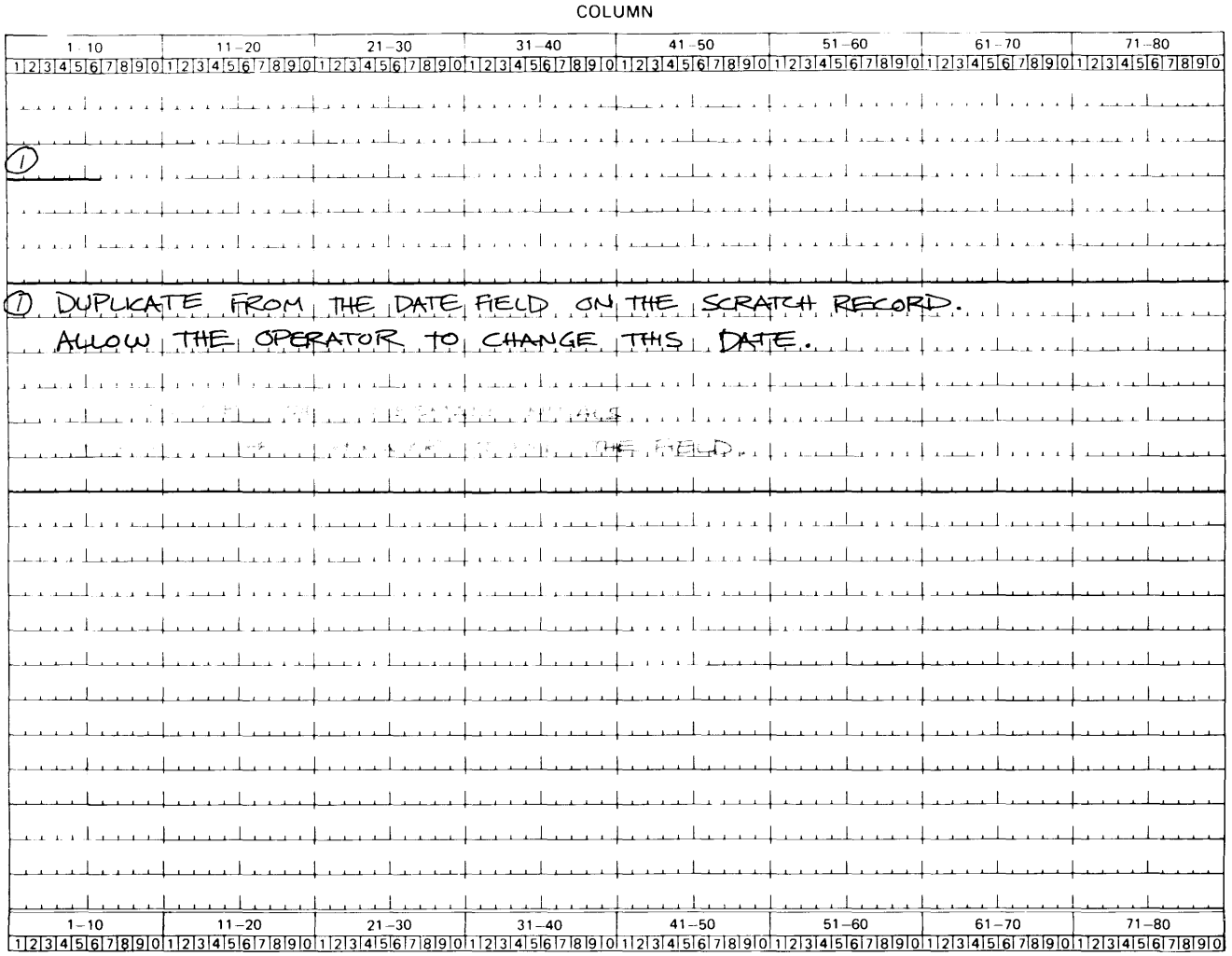
The first field is the date field. It is duplicated from the date field in the scratch record. According to the instructions for this job, this field is to be changed if the date on a sales order form changes.

Display Screen Layout Sheet



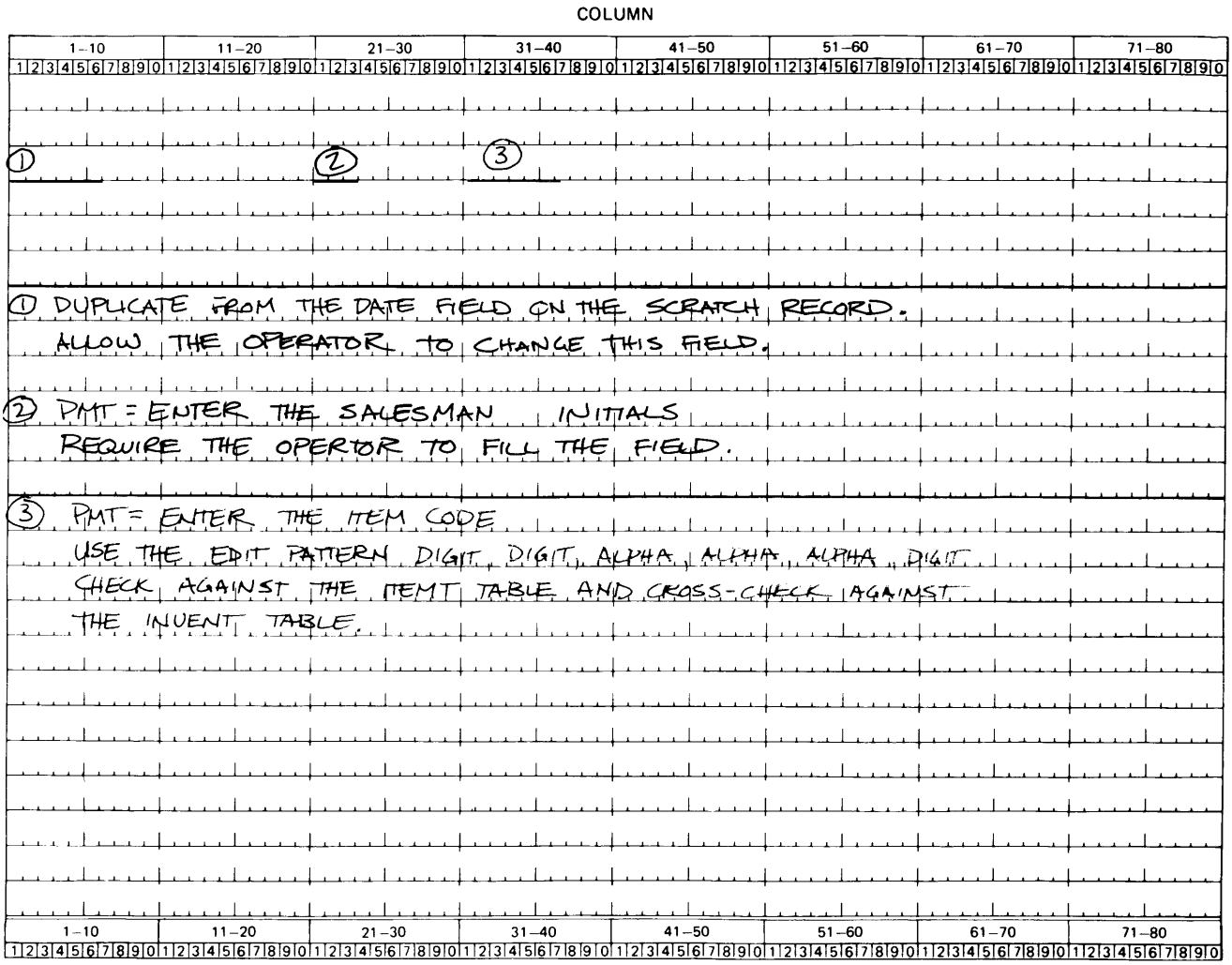
The next field is for the salesman's initials. It does not contain any functions that have not previously been used.

Display Screen Layout Sheet



Now, look at the item field. The instructions limit the entry in this field to a pattern: digit, digit, alphabetic, alphabetic, alphabetic, digit. This patterning is sometimes called a picture check. If the entry matches the pattern, the operator is allowed to continue. If it does not match, an error occurs. The item field also includes a prompt and uses a table function.

Display Screen Layout Sheet



The table function that is used looks in the table that is named (ITEMT) to see whether the item number that was entered exists. A cross-check table function involving this field and the price field is also specified.

The description field is next; it provides space for descriptive information about the item. There is no restriction on this field. You should, however, include a prompt to guide the operator.

Display Screen Layout Sheet

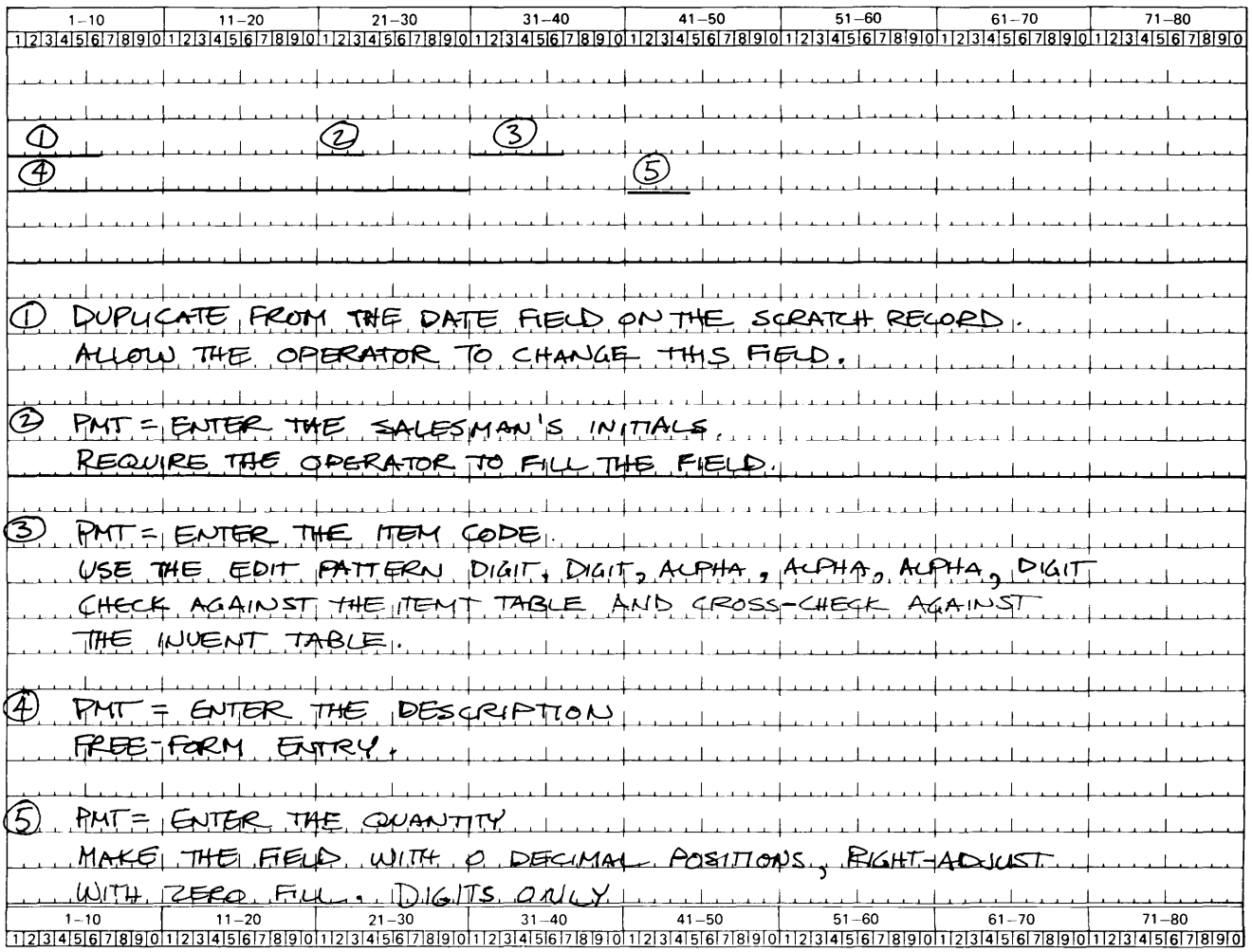
COLUMN

1–10	11–20	21–30	31–40	41–50	51–60	61–70	71–80
1 2 3 4 5 6 7 8 9 0	1 2 3 4 5 6 7 8 9 0	1 2 3 4 5 6 7 8 9 0	1 2 3 4 5 6 7 8 9 0	1 2 3 4 5 6 7 8 9 0	1 2 3 4 5 6 7 8 9 0	1 2 3 4 5 6 7 8 9 0	1 2 3 4 5 6 7 8 9 0
①		②	③				
④							
① DUPLICATE FROM THE DATE FIELD ON THE SCRATCH RECORD. ALLOW THE OPERATOR TO CHANGE THIS FIELD.							
② PMT = ENTER THE SALESMAN INITIALS REQUIRE THE OPERATOR TO FILL THE FIELD.							
③ PMT = ENTER THE ITEM CODE. USE THE EDIT PATTERN DIGIT, DIGIT, ALPHA, ALPHA, ALPHA, DIGIT CHECK AGAINST THE ITEMT TABLE AND CROSS-CHECK AGAINST THE INUENT TABLE.							
④ PMT = ENTER THE DESCRIPTION FREE-FORM ENTRY.							
1–10	11–20	21–30	31–40	41–50	51–60	61–70	71–80
1 2 3 4 5 6 7 8 9 0	1 2 3 4 5 6 7 8 9 0	1 2 3 4 5 6 7 8 9 0	1 2 3 4 5 6 7 8 9 0	1 2 3 4 5 6 7 8 9 0	1 2 3 4 5 6 7 8 9 0	1 2 3 4 5 6 7 8 9 0	1 2 3 4 5 6 7 8 9 0

The quantity field is next. It specifies that the entry is to be right-adjusted and the remaining unused field positions filled with zeros. Another instruction requires that this field should contain digits only and should be specified as having zero decimal positions. A field without an entry in the decimal position columns of the A specification cannot be used in calculations. You will want to use this field to determine the cost of the items, so it must have a decimal position entry (even if the entry is a 0) in order to be used in a calculation.

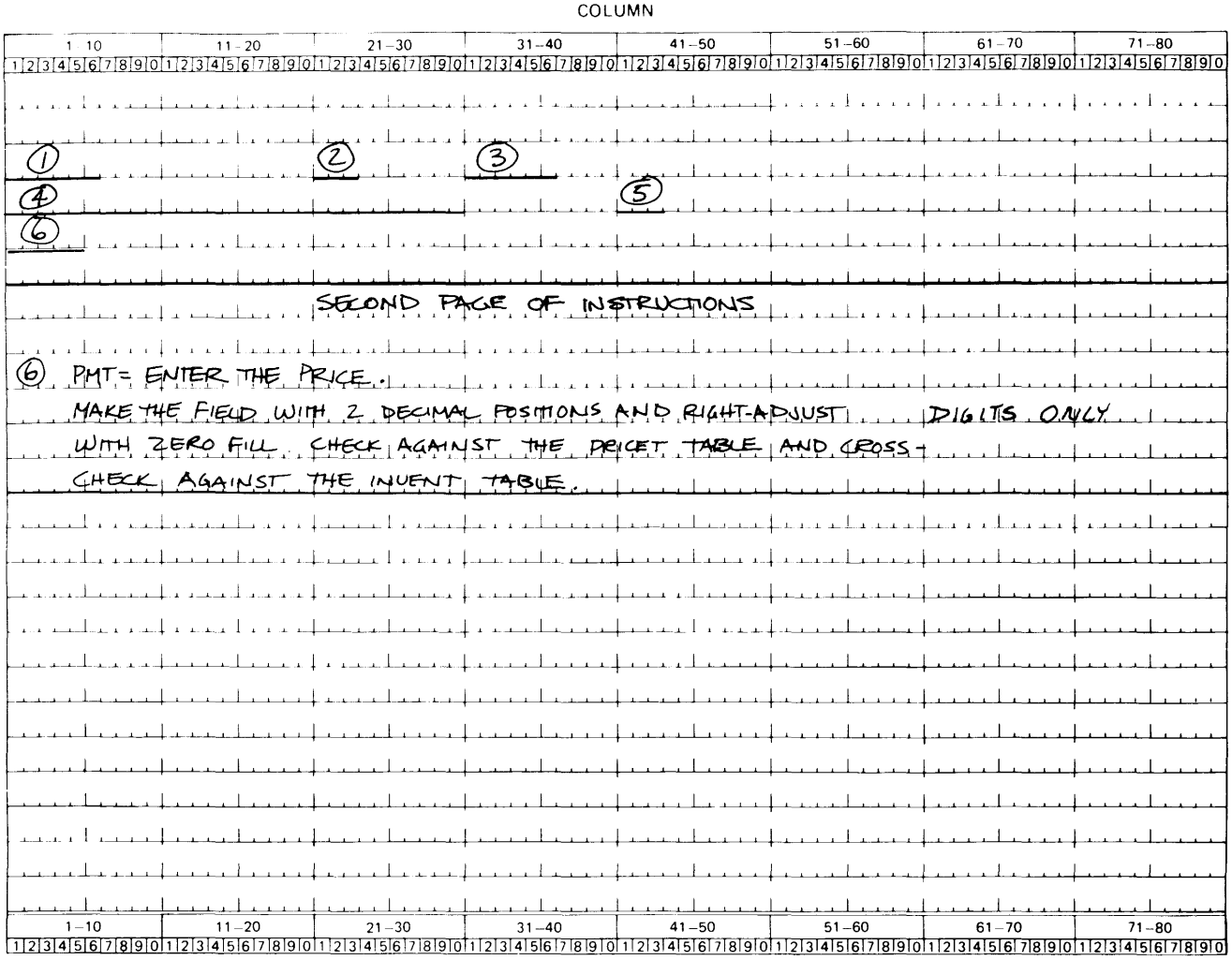
Display Screen Layout Sheet

COLUMN



The price field is next. This field is right-adjusted with zero fill and allows digit-only entries. Two decimal positions are specified. In addition, the entry in the price field is specified as being checked against the PRICET table to verify that the business sells an item at the price entered. This field is also specified as being used in a cross-check table function along with the item field. The INVENT table is used to perform the cross-check.

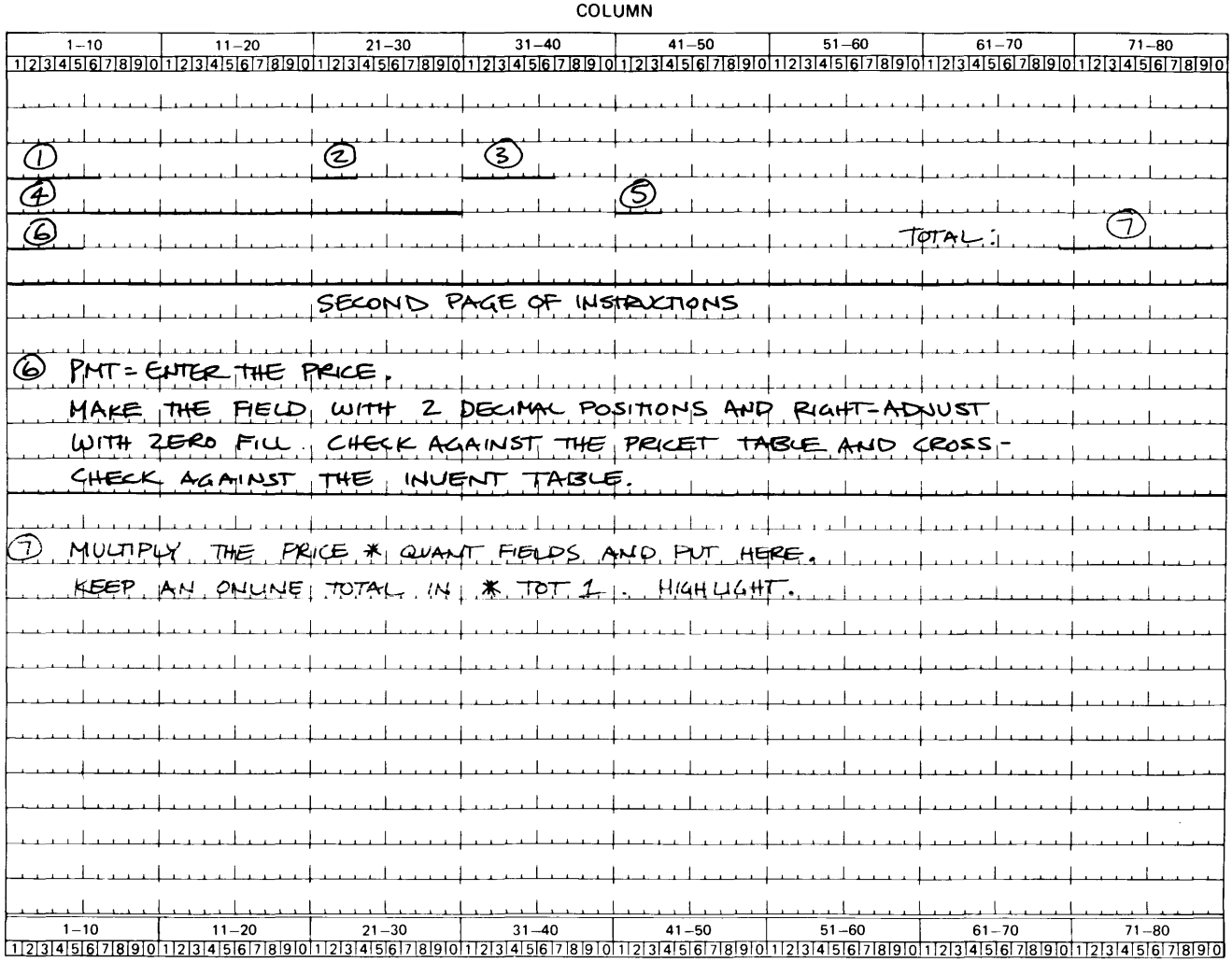
Display Screen Layout Sheet



The next field is a simple literal (TOTAL:). It can be automatically supplied by DE/RPG.

The cost field is the actual entry for the price of the items. DE/RPG automatically supplies the contents of the field by multiplying the named fields PRICE and QUANT. Directions for maintaining an online total in the *TOT1 counter are also specified. *TOT1 is the counter you were told to use in the instructions from the supervisor. Notice that the field is marked to be highlighted. Automatically supplied fields are provided so rapidly that it is helpful for the operator to have some means of identifying when they are complete. Highlighting should provide this means.

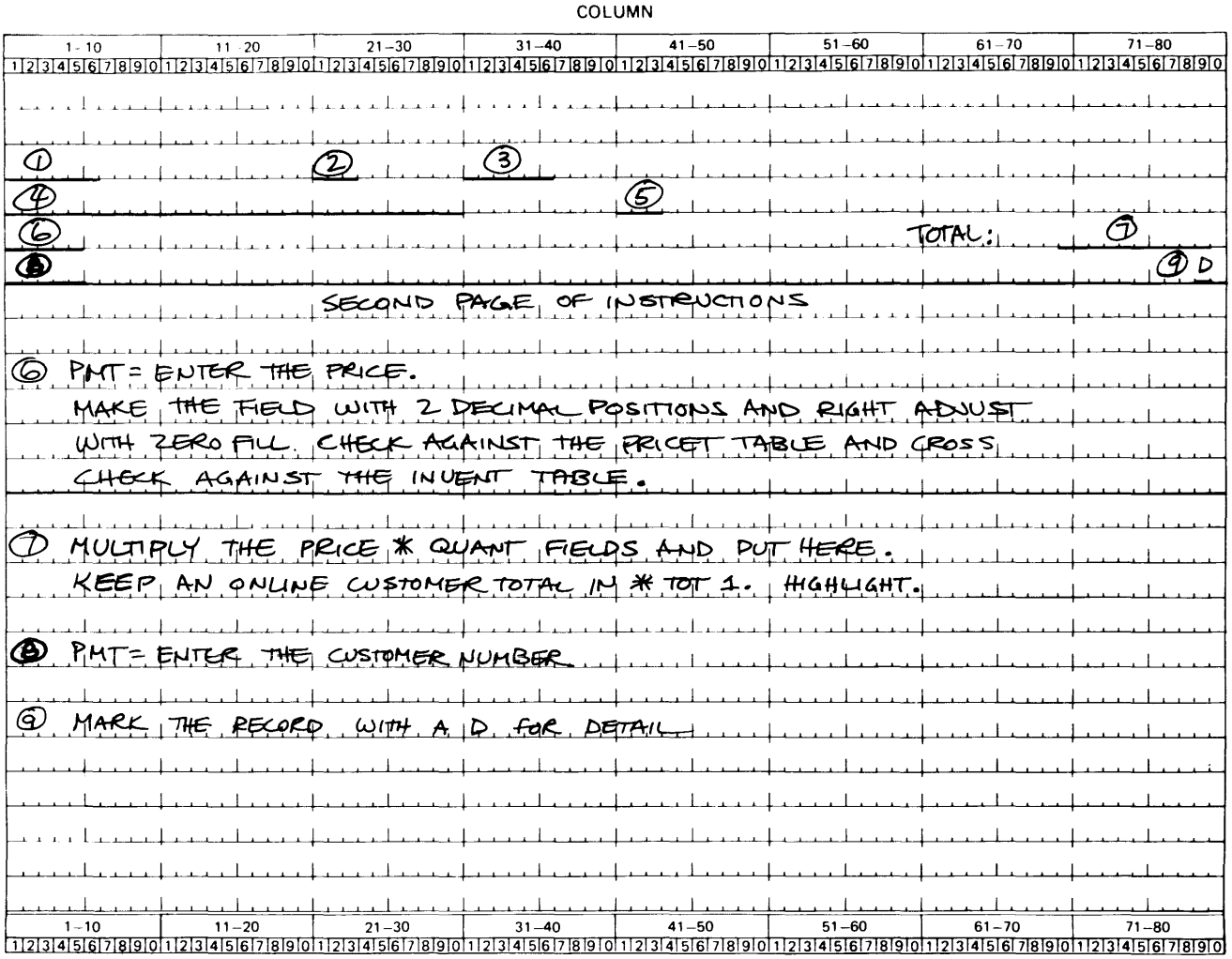
Display Screen Layout Sheet



The next field is the customer number field. Mark the field to prompt the operator for the customer number.

The last field is the record type (D) mark field. The D is automatically inserted to provide the record marker for the detail record.

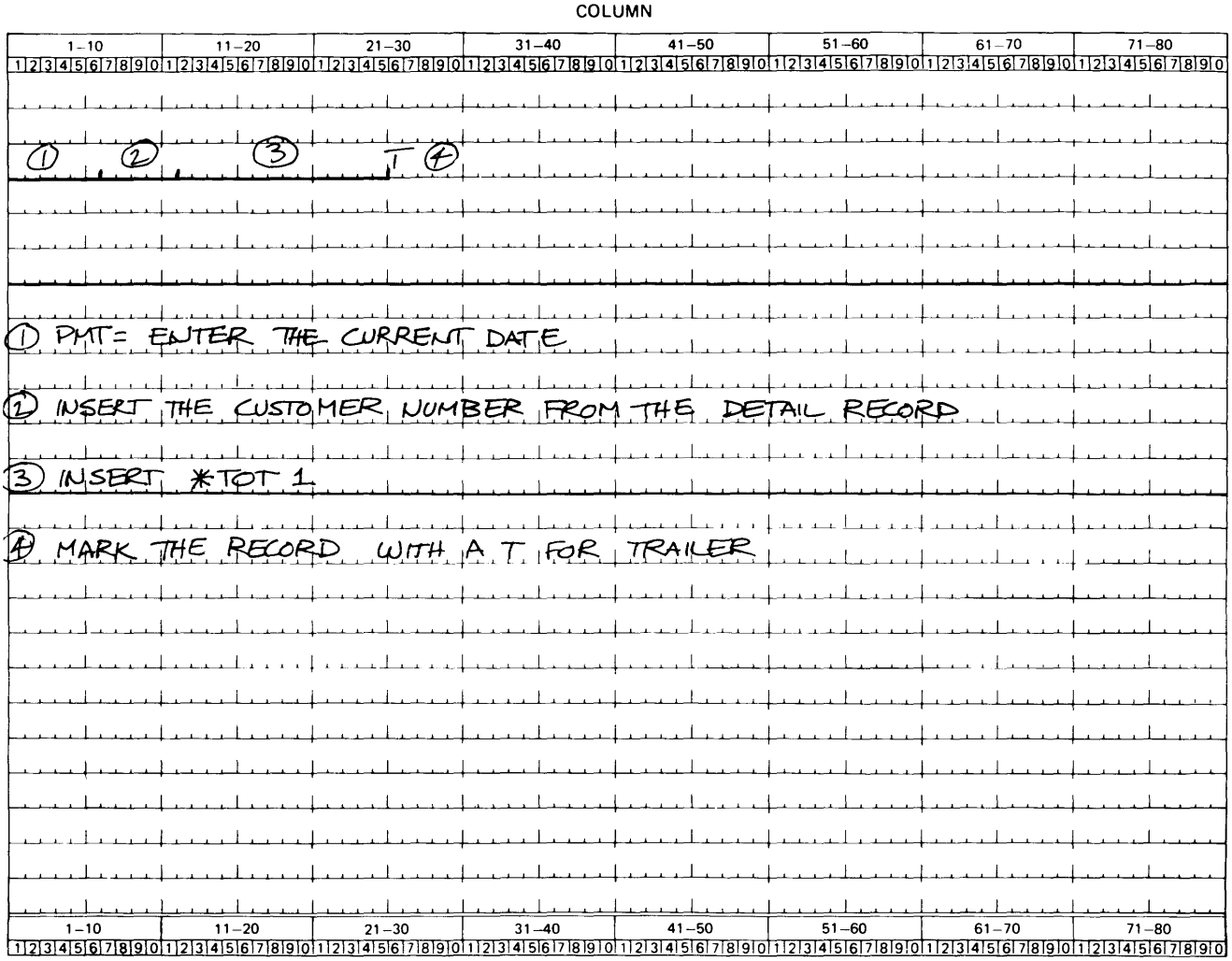
Display Screen Layout Sheet



You have completed the detail display record description on the work sheets. Before you code this description on the A specification, complete the field descriptions for the trailer (TRAIL) display record.

Most fields in the trailer display record are automatically supplied by DE/RPG; that is, they are not entered by the operator. The first field is the date field. It should not be automatically duplicated from a previous scratch or detail record because this field contains the current date, which probably does not match the date on the orders. This is the only field in the trailer record that will require entry.

Display Screen Layout Sheet



The second field is for the customer number; it can be automatically duplicated from the customer number field in the detail record.

The third field is obtained by inserting the value of *TOT1, which has been accumulated for all the detail records. When you understand how the detail and trailer records are used, you will understand how *TOT1 can be used to keep an online total.

The last field is the marker for the trailer record. Use the insert function to place a T in each trailer record just as the H was placed in the header record earlier. The *DE/RPG Reference Manual* describes the insert function in detail.

You have completed the display field descriptions for the scratch, detail, and trailer records which, along with the header record from the first job, were the records specified to be included in the data set for this sample job.

HOW THE OPERATOR WILL USE THE MASTHEAD DATA SET AND THE DETAILED PURCHASE JOB

It is difficult to write the detailed purchase program without understanding how it will be used. Remember that the first job created a master data set containing general identification information about each customer. The second job, the one for which you have designed these displays, will copy information from this master data set as needed. Copying will be accomplished by use of the keyword CFILE in the job description statement of the Z specification and by the use of the copy keys.

In use, the program will work in the following way. The operator will see these words: USE THE SEARCH KEYS TO FIND THE CORRECT HEADER RECORD IN THE MASTHEAD DATA SET. SEARCH FOR A MATCH TO THE CUSTOMER NUMBER ON THE ORDER FORM. NEXT USE THE COPY KEYS TO COPY THE DATA FROM THE HEADER RECORD INTO THIS DATA SET.

The operator will search the MASTHEAD data set for a match to the customer number; then the operator will use the keyboard to perform the copy function. The header record containing the customer number that matches the sales order form will be written on the diskette. The data in the header record will be displayed in format 0. Format 0 is the default format for DE/RPG; each character in format 0 is considered a field.

Next, DE/RPG will display the format for entering the detailed information. The operator will continue using the detailed record format until all purchase information for the customer has been entered.

When the operator selects the next format from the keyboard, the format for the trailer record will be displayed and the automatic total functions will be performed and the result will be written on diskette. The format containing the prompt that tells the operator to use the copy function will automatically be displayed again. The operator will again search the MASTHEAD data set for the next customer number that matches the next set of sales order forms and then copy the matching record into the BILLING data set.

This sequence of a header record, one or more detail records, and a trailer record will continue until the operator enters all the information from the sales order forms and uses the EOJ function on the keyboard to end this job.

DESCRIBING THE DISPLAY FILE, RECORDS, AND FIELDS ON THE A SPECIFICATION

In the first job, when you were learning how to write a program, you started the sample program by defining the fields first.

This time, you will start at the normal beginning of the program, with the file description for the display (CRT) on the A specification.

As in the first sample, the second sample requires a file description for the display.

IBM International Business Machines Corporation — IBM 5280 DATA DESCRIPTION SPECIFICATIONS — Printed in U.S.A.

Job No. BILLING	Dataset	Keying Instruction	Graphic	Source Document DETAILED PURCHASE JOB	Page 2	of 8
Operator	Date		Key			

A

Sequence	Form Type	Comment (*)	Reserved	Indicator (for CHECK (BY, BV) or ERROR)	Reserved	Name Type (F/K/R/T)	Reserved	Dataset/Record/ Field/Table Name	Reserved	Length	Data Type	Reserved	Decimal Positions (0-9)	Usage (I/O/B/W)	Location: Screen Line	Location: Screen Pos	Editing
1	A					F		DTSP		150							DEVICE(CRT)
2	A																DSPSIZ(6 80)
3	A																
4	A																
5	A																
6	A																
7	A																
8	A																
9	A																
10	A																
11	A																
12	A																
13	A																
14	A																
15	A																
	A																
	A																
	A																
	A																
	A																

Checks=CHECK (code . . .)

Auto Dup	+AD	Mandatory Entry	+ME
Auto Skip	+AS	Mandatory Fill	+MF
Blank Check	+BC	Rt Adj–Blank Fill	+RB
Bypass	+BY	Right to Left	+RL
Bypass on Verify	+BV	Rt Adj–Zero Fill	+RZ
Data Required	+DR	Self-Check	+nnn
Dup Disable	+DD	n=C,G (Check Gen)	
Field Exit Required	+FE	xx=Modulus	
Lower Case	+LC		

Functions

ADD (name)	PMT (prompt)
AUXDUP (name)	RANGE (low high)
AUXST (name)	RANGET (table [index])
COMP (¹test fld1²@ fld2 [indicator])	RESET (+TOTn)
DSPATR (²attr)	SEQ (¹test)
EDTCDE (code [float])	SETOF (ind)
ERROR (code [³message])	SETON (ind)
EXSR (subroutine)	SHIFT (⁴shift)
INSERT (fld1²@ fld2)	SUB (name)
LOOK (table [index])	SUBST (table1 table2 [index])
	TADD (+TOTn)
¹test=EQ,GE,GT,LE,LT,NE	TSUB (+TOTn)
²attr=BL,CA,CS,HI,ND,RI,UL	XCHK (table index1 index2)
³@=*,=,+,-	literal
⁴shift=A,D,H,N,V,W,X,Y	

*Number of sheets per pad may vary slightly

Following the file description, you must define the first record. The first record you defined on the work sheet is the scratch record. This record describes a special display that allows you to prompt the operator for a date and provide information describing the use of the job. This record was not specified in the instructions for the job; therefore, it will not be included in the diskette data set. Actually, the use of the scratch record is a programming technique that you can use in a variety of applications. It allows you to set up fields that you can duplicate or insert in other records and provides extra prompting which precedes another record.

Describing the Scratch Record Using the A Specification

You define this record as you did the records for the first job. The format description on the Z specification will determine whether or not it is written on the diskette. The description on the A specification is the same as all other records. This record has three different kinds of fields: (1) a date field, (2) fields for prompting (literals), and (3) a field for operator control in exiting the display.

IBM International Business Machines Corporation — IBM 5280 DATA DESCRIPTION SPECIFICATIONS — Printed in U.S.A.

Job No. BILLING	Dataset	Keying Instruction	Graphic		Source Document DETAILED PURCHASE JOB	Page 2 of 8
Operator	Date		Key			

A

Sequence	Form Type	Comment (*)	Reserved	Indicator (for CHECK (BY, BV) or ERROR)	Reserved	Name Type (F/K/R/T)	Reserved	Dataset/Record/ Field/Table Name	Reserved	Length	Data Type	Reserved	Decimal Positions (0-9)	Usage (I/O/B/W)	Location: Screen Line	Location: Screen Pos	Editing: Checks=CHECK (code . . .) / Functions
1	A					F		DISP		150							DEVICE(CRT)
2	A																DSPSIZ(6 80)
3	A					R		SCRATCH									
4	A									6				I	00	10	71PMT(USE AUTODUP TO ENTER NEW DATE O+
5	A																R CHANGE DATE)
6	A																AUXST(DATE)
7	A																
8	A																
9	A																
10	A																
11	A																
12	A																
13	A																
14	A																
15	A																
	A																
	A																
	A																
	A																
	A																

Editing — Checks=CHECK (code . . .)

Auto Dup	AD	Mandatory Entry	ME
Auto Skip	AS	Mandatory Fill	MF
Blank Check	BC	Rt Adj–Blank Fill	RB
Bypass	BY	Right to Left	RL
Bypass on Verify	BV	Rt Adj–Zero Fill	RZ
Data Required	DR	Self Check	mxx
Dup Unsafe	DU	n=1 to 9 check term	
Field Exit Required	FE	xx Modulus	
Lower Case	LC		

Editing — Functions

ADD (name)	PMT (prompt)
AUXDUP (name)	RANGE (low high)
AUXST (name)	RANGET (table [index])
COMP (1test fld1 1@ fldn indicator;)	RESET (+TOTn)
DSPATR (1attr)	SEQ (1test)
EDTCDE (code float)	SETOF (ind)
EMHUM (code [,message])	SETON (ind)
EXSR (subroutine)	SHIFT (4shift)
INSERT (fld1 1@ fldn)	SUB (name)
LOOK (table [,index])	SUBST (table1 table2 [index])
	TADD (+TOTn)
1test=EQ,GE,GT,LE,LT,NE	TSUB (+TOTn)
2attr=BL,CA,CS,HI,ND,RI,UL	XCHK (table index1 index2)
3@=+ , - , *	4literal
4shift=A,D,H,N,V,W,X,Y	

1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20 21 22 23 24 25 26 27 28 29 30 31 32 33 34 35 36 37 38 39 40 41 42 43 44 45 46 47 48 49 50 51 52 53 54 55 56 57 58 59 60 61 62 63 64 65 66 67 68 69 70 71 72 73 74 75 76 77 78 79 80

*Number of sheets per pad may vary slightly.

The prompt for the date field guides the operator in identifying when to enter information in this field. Notice that the date field uses the keyword and parameter AUXST(DATE). Whenever the Auto Dup keyboard function is active and the operator enters data in the field, the entry is stored in the name DATE. Later, if another field uses the DATE field (for example, when AUXDUP(DATE) is used in the DET record), the value of the original entry is placed in the field. Using the AUXST keyword allows the operator to change the value of DATE easily but only when necessary. Whenever the Auto Dup function is inactive, the operator can change the value of DATE.

The next fields in the scratch record provide the operator prompting. Notice on the A specification that the messages have been designed to fit into 80 columns.

IBM International Business Machines Corporation — IBM 5280 DATA DESCRIPTION SPECIFICATIONS — Printed in U.S.A.

Job No. BILLING	Dataset	Keying Instruction	Graphic								Source Document DETAILED PURCHASE JOB	Page 2 of 8
Operator	Date		Key									

A

Sequence	Form Type	Comment (*)	Reserved	Indicator (for CHECK (BY, BV) or ERROR)	Reserved	Name Type (F/K/R/T)	Reserved	Dataset/Record/ Field/Table Name	Reserved	Length	Data Type	Reserved	Decimal Positions (0-9)	Usage (I/O/B/W)	Location Screen Line	Location Screen Pos	Editing
1	A					F		DISP		150							DEVICE(CRT)
2	A																DSPSIZ(6 80)
3	A					R		SCRATCH									
4	A									6				I	00	107	1PMT(USE AUTODUP TO ENTER NEW DATE O+
5	A																R CHANGE DATE)
6	A																AUXST(DATE)
7	A														00	2001	'USE THE SEARCH KEYS TO FIND THE CO+
8	A																RRECT HEADER RECORD IN THE MASTHEAD-
9	A																DATA SET.'
10	A														00	3001	'SEARCH FOR A MATCH TO THE CUSTOMER-
11	A																NUMBER ON THE ORDER FORM.'
12	A														00	4001	'NEXT USE THE COPY KEYS TO COPY DAT+
13	A																A FROM THE HEADER INTO THIS DATA SE+
14	A																T.'
15	A									1				I			PMT(USE THE FIELD EXIT KEY TO LEAVE-
	A																THE DISPLAY)
	A																CHECK(FE)
	A																
	A																
	A																
	A																

Checks=CHECK (code . . .)

Auto Dup	=AD	Mandatory Entry	=ME
Auto Skip	=AS	Mandatory Fill	=MF
Blank Check	=BC	Rt Adj–Blank Fill	=RB
Bypass	=BY	Right to Left	=RL
Bypass on Verify	=BV	Rt Adj–Zero Fill	=RZ
Data Required	=DR	Self Check	=nnx
Dup Disable	=DD	n=C/G (Check/Gen)	
Field Exit Required	=FE	xx=Modulus	
Lower Case	=LC		

Functions

ADD (name)	PMT (prompt)
AUXDUP (name)	RANGE (low high)
AUXST (name)	RANGET (table [index])
COMP ([1]test fld[2]@ fldn [indicator])	RESET (+TOTn)
DSPATR ([2]attr)	SEQ ([1]test)
EDTCDE (code [float])	SETOF (ind)
ERROR (code ['message'])	SETON (ind)
EXSR (subroutine)	SHIFT ([4]shift)
INSERT (fld[3]@ fldn)	SUB (name)
LOOK (table [index])	SUBST (table1 table2 [index])
	TADD (+TOTn)
[1]test=EQ,GE,GT,LE,LT,NE	TSUB (+TOTn)
[2]attr=BL,CA,CS,HI,ND,RI,UL	XCHK (table index1 index2)
[3]@=+, -, *, /	literal
[4]shift=A,D,H,N,V,W,X,Y	

1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20 21 22 23 24 25 26 27 28 29 30 31 32 33 34 35 36 37 38 39 40 41 42 43 44 45 46 47 48 49 50 51 52 53 54 55 56 57 58 59 60 61 62 63 64 65 66 67 68 69 70 71 72 73 74 75 76 77 78 79 80

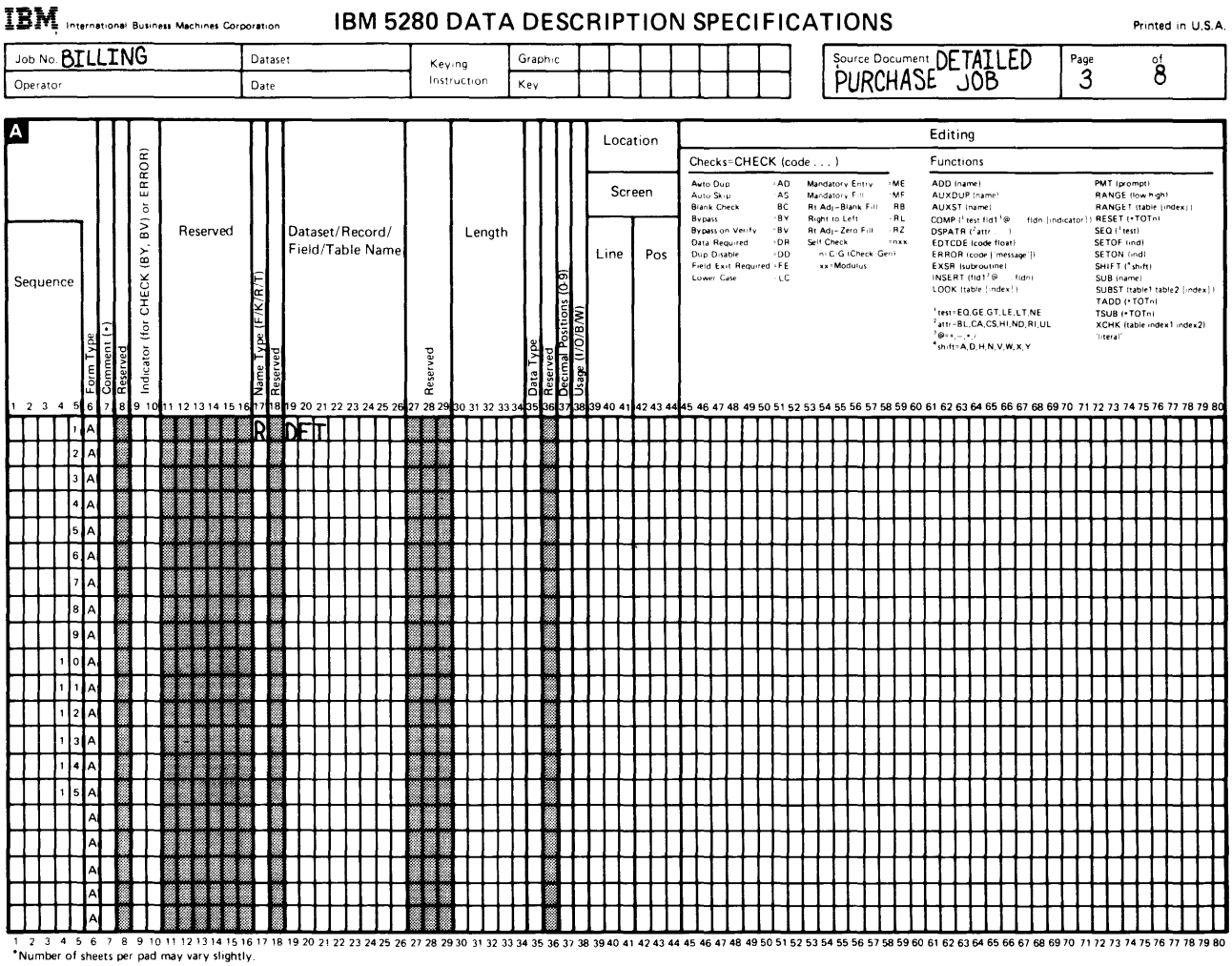
*Number of sheets per pad may vary slightly.

A single-position field follows the instructions. The reason for including this field is that the prompts will be displayed quickly and without operator control. The operator needs to see the display and read its contents to perform the actions necessary to copy the header record from the MASTHEAD data set. To do this, you must halt the display. The single-position field with the CHECK(FE) keyword and parameter allows you to do this. The prompt explains that the operator must use a field exit key to leave the display. This allows the operator to determine when to exit from the display. When there are no entries in the location columns of the A specification, that field is placed immediately after the preceding field. Therefore, this single-spaced field will be immediately after the period following the word SET in the preceding literal.

The definition for the scratch record is complete. You are ready to define the next record.

Describing the DET (Detail Display) Record Using the A Specification

The record description line is the same as for the scratch record except the name is now DET. The sample shows this record description beginning on a new A specification. This is not necessary; it is used in the samples to help you see the record contents clearly. If you are confused because the name of the record is not DETAIL, remember that a reformatted diskette record cannot have the same name as the display record, so you should reserve the name DETAIL for the record that is written on the diskette.



The first field is for the date. Fields in different records of the same program should have unique names to avoid confusion; because a field named DATE already exists in the scratch record, name this field DAT. Look at the supervisor's instructions and the display work sheet to determine the length and position of the field. The prompt is written in the same way as the prompts for the first sample program were written. The only new concept is the use of the AUXDUP keyword to duplicate the contents of the Date field from the scratch record. Whenever AUXDUP(DATE) is specified, the current contents of DATE are automatically inserted into the current field (DAT) if the Dup key is pressed.

IBM International Business Machines Corporation

IBM 5280 DATA DESCRIPTION SPECIFICATIONS

Printed in U.S.A.

Job No. BILLING	Dataset	Keying Instruction	Graphic	Source Document DETAILED PURCHASE JOB	Page 3	of 8
Operator	Date		Key			

A

Sequence | Form Type | Comment (*) | Reserved | Indicator (for CHECK (BY, BV) or ERROR) | Reserved | Name Type (F/K/R/T) | Reserved | Dataset/Record/Field/Table Name | Reserved | Length | Data Type | Reserved | Decimal Positions (0-9) | Usage (I/O/B/W) | Location: Line, Pos | Editing

Checks=CHECK (code . . .)

Auto Dup	AD	Mandatory Entry	ME
Auto Skip	AS	Mandatory Fill	MF
Blank Check	BC	Rt Adj–Blank Fill	RB
Bypass	BY	Right to Left	RL
Bypass on Verify	BV	Rt Adj–Zero Fill	RZ
Data Required	DR	Self Check	nnxx
Dup Disable	DD	n–C/G (Check/Gen)	
Field Exit Required	FE	xx–Modulus	
Lower Case	LC		

Functions

ADD (name)	PMT (prompt)
AUXDUP (name)	RANGE (low high)
AUXST (name)	RANGET (table (index))
COMP ([1]test fld1[2]@ fldn [indicator])	RESET (+TOTn)
DSPATR ([2]attr)	SEQ ([1]test)
EDTCDE (code float)	SETOF (ind)
ERROR (code ['message'])	SETON (ind)
EXSR (subroutine)	SHIFT ([4]shift)
INSERT (fld1[3]@ fldn)	SUB (name)
LOOK (table (index))	SUBST (table1 table2 (index))
	TADD (+TOTn)
[1]test=EQ,GE,GT,LE,LT,NE	TSUB (+TOTn)
[2]attr=BL,CA,CS,HI,ND,RI,UL	XCHK (table index1 index2)
[3]@=+ - * /	'literal'
[4]shift=A,D,H,N,V,W,X,Y	

Sequence	Form Type	Name Type	Dataset/Record/Field/Table Name	Length	Data Type	Decimal Positions	Usage	Line	Pos	Editing
10	A	R	DET							
20	A		DAT	6			I	00	01	AUXDUP(DATE)
30	A									
40	A									
50	A									
60	A									
70	A									
80	A									
90	A									
100	A									
110	A									
120	A									
130	A									
140	A									
150	A									
	A									
	A									
	A									
	A									
	A									

*Number of sheets per pad may vary slightly.

The next field is the salesman's initials field. There are no new concepts.

IBM International Business Machines Corporation

IBM 5280 DATA DESCRIPTION SPECIFICATIONS

Printed in U.S.A.

Job No. BILLING	Dataset	Keying Instruction	Graphic								Source Document DETAILED PURCHASE JOB	Page 3	of 8
Operator	Date		Key										

A

Sequence	Form Type	Comment (*)	Reserved	Indicator (for CHECK (BY, BV) or ERROR)	Reserved	Name Type (F/K/R/T)	Reserved	Dataset/Record/ Field/Table Name	Reserved	Length	Data Type	Reserved	Decimal Positions (0-9)	Usage (I/O/B/W)	Location Screen Line	Location Screen Pos	Editing
1	A					R		DET									
2	A							DAT			6			I	002	001	AUXDUP(DATE)
3	A							SALS			3X			I	002	021	PMT(ENTER THE SALESMAN INITIALS)
4	A																CHECK(DR BC)
5	A																
6	A																
7	A																
8	A																
9	A																
10	A																
11	A																
12	A																
13	A																
14	A																
15	A																
	A																
	A																
	A																
	A																
	A																

Checks=CHECK (code,...)

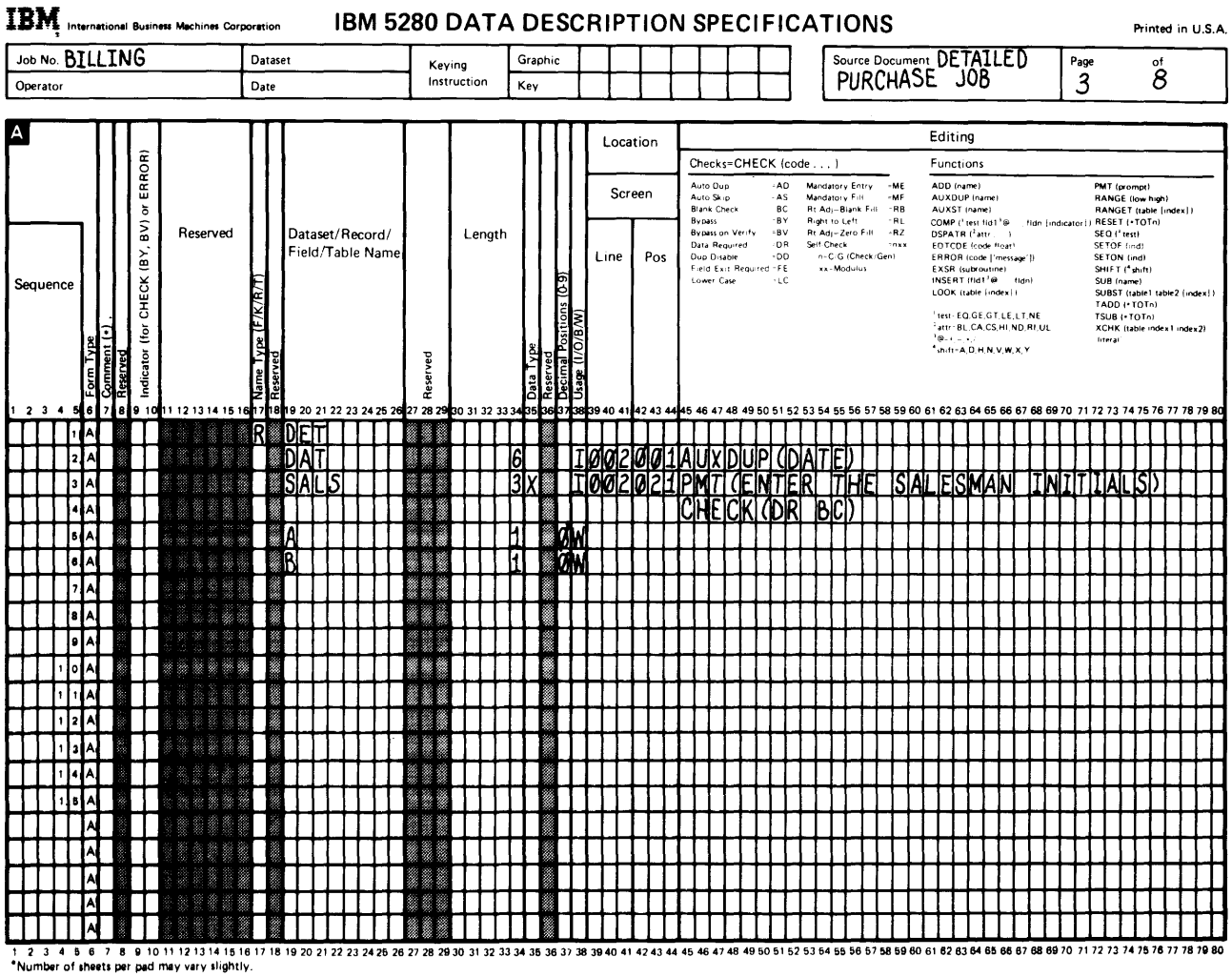
Auto Dup	AU	Mandatory Entry	ME
Auto Skip	AS	Mandatory Fill	MF
Blank Check	BC	Rt Adj, Blank Fill	RB
Bypass	BY	Right to Left	RL
Bypass on Verify	BV	Rt Adj, Zero Fill	RZ
Data Required	DR	Self Check	**
Dup Shadow	DD	C,G (Check Digit)	
Field Exit Required	FE	** Modulus	
Lower Case	LC		

Functions

ADD (name)	PMT (prompt)
AUXDUP (name)	RANGE (low high)
AUXKST (name)	RANGET (table [index])
COMP (test tid1 tid	RESET (+TOTn)
DSPATR (attr	SEQ (test)
EDTCDE (code [loat)	SETOF (ind)
ERROR (code [,message])	SETON (ind)
EXSR (subroutine)	SHIFT (*shift)
INSERT (tid1 tid	SUB (name)
LOOK (table [,index])	SUBST (table1 table2 [,index])
	TADD (+TOTn)
test: EQ,GE,GT,LE,LT,NE	TSUB (+TOTn)
attr: BL,CA,CS,HI,ND,RI,UL	XCHK (table,index1,index2)
@	'literal
*shift: A,D,H,N,V,W,X,Y	

*Number of sheets per pad may vary slightly.

Fields that are named A and B follow the salesman's initials field. Fields A and B are the indexes you were instructed to use for the table functions. Notice that the length of each field is 1. Because only six entries are in the ITEMT and PRICET tables, a single-position index is sufficient. Also, notice that the usage column contains a W (work space). The W prevents the field from being displayed, altered, or written on the diskette. The zero in the decimal position column describes the fields as numeric. Whenever indexes are used, they must exist as numeric fields.



IBM International Business Machines Corporation — IBM 5280 DATA DESCRIPTION SPECIFICATIONS — Printed in U.S.A.

Job No. BILLING | Dataset | Keying Instruction | Graphic | Key
Operator | Date

Source Document DETAILED PURCHASE JOB | Page 3 of 8

Sequence	Form Type	Comment (*)	Reserved	Indicator (for CHECK (BY, BV) or ERROR)	Reserved	Name Type (F/K/R/T)	Reserved	Dataset/Record/ Field/Table Name	Reserved	Length	Data Type	Reserved	Decimal Positions (0-9)	Usage (I/O/B/W)	Location: Line	Location: Pos	Editing
1	A					R		DET									
2	A							DAT		6				I	00	20	01AUXDUP(DATE)
3	A							SALS		3	X			I	00	20	21PMT(ENTER THE SALESMAN INITIALS)
4	A																CHECK(DR BC)
5	A							A		1			0	W			
6	A							B		1			0	W			
7	A																
8	A																
9	A																
10	A																
11	A																
12	A																
13	A																
14	A																
15	A																
	A																
	A																
	A																
	A																
	A																

Checks=CHECK (code . . .)

Auto Dup	-AD	Mandatory Entry	-ME
Auto Skip	-AS	Mandatory Fill	-MF
Blank Check	-BC	Rt Adj-Blank Fill	-RB
Bypass	-BY	Right to Left	-RL
Bypass on Verify	-BV	Rt Adj-Zero Fill	-RZ
Data Required	-DR	Self-Check	-nnx
Dup Disable	-DD	n-C,G (Check (Gen)	
Field Exit Required	-FE	xx-Modulus	
Lower Case	-LC		

Functions

ADD (name)
AUXDUP (name)
AUXST (name)
COMP ('test fld1'@ fldn [indicator])
DSPATR ('attr')
EDTCDE (code [mask])
ERROR (code ['message'])
EXSR (subroutine)
INSERT (fld1'@ fldn)
LOOK (table [index])

PMT (prompt)
RANGE (low high)
RANGET (table [index])
RESET (+TOTn)
SEQ ('test)
SETOF (ind)
SETON (ind)
SHIFT ('shift)
SUB (name)
SUBST (table1 table2 (index))
TADD (+TOTn)
TSUB (+TOTn)
XCHK (table index1 index2)

'test - EQ,GE,GT,LE,LT,NE
'attr - BL,CA,CS,HI,ND,RI,UL
'@ - + -
'shift - A,D,H,N,V,W,X,Y

literal

1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20 21 22 23 24 25 26 27 28 29 30 31 32 33 34 35 36 37 38 39 40 41 42 43 44 45 46 47 48 49 50 51 52 53 54 55 56 57 58 59 60 61 62 63 64 65 66 67 68 69 70 71 72 73 74 75 76 77 78 79 80

*Number of sheets per pad may vary slightly.

The item number field is next. This field contains the picture check pattern for entry. The picture check has two parts: (1) a C must be in the data type column and (2) the SHIFT keyword and the parameter that forms the pattern for the picture check must be used. For this field, the parameter DDXXXD means that the first two entries must be digits, the next three must be alphabetic characters, and the last must be a digit.

The CHECK (DR) keyword and parameter are used for this field also. To exit the field, the operator must use one of the field exit keys.

Now, look at the next keyword and its parameters. This is the first table function that you will use. The LOOK keyword describes the function. The LOOK keyword *looks* in the table named as its parameter to see whether the entry in the current field matches one of the entries in the table. Notice that an A follows the table name. This A is the index used for the table function. You only need to use indexes with the tables because you want to cross-check the table with another table. You know that this entry is to be cross-checked with the price field, so use an index.

IBM International Business Machines Corporation

IBM 5280 DATA DESCRIPTION SPECIFICATIONS

Printed in U.S.A.

Job No. BILLING	Dataset	Keying Instruction	Graphic	Source Document DETAILED PURCHASE JOB	Page 3 of 8
Operator	Date		Key		

A

Sequence	Form Type	Comment (*)	Reserved	Indicator (for CHECK (BY, BV) or ERROR)	Reserved	Name Type (F/K/R/T)	Reserved	Dataset/Record/ Field/Table Name	Reserved	Length	Data Type	Reserved	Decimal Positions (0-9)	Usage (I/O/B/W)	Location: Screen Line	Location: Screen Pos	Editing: Checks=CHECK (code . . .) / Functions
1	A					R		DET									
2	A							DAT			6			I	00	20	01AUXDUP(DATE)
3	A							SALS			3X			I	00	20	21PMT(ENTER THE SALESMAN INITIALS)
4	A																CHECK(DR BC)
5	A							A			1		0	W			
6	A							B			1		0	W			
7	A							ITEM			6C			I	00	20	31PMT(ENTER THE ITEM CODE)
8	A																SHIFT(DDXXXD)
9	A																CHECK(DR)
10	A																LOOK(ITEMT A)
11	A																
12	A																
13	A																
14	A																
15	A																
	A																
	A																
	A																
	A																
	A																
	A																

Editing — Checks=CHECK (code . . .)

Auto Dup	=AD	Mandatory Entry	=ME
Auto Skip	=AS	Mandatory Fill	=MF
Blank Check	=BC	Rt Adj–Blank Fill	=RB
Bypass	=BY	Right to Left	=RL
Bypass on Verify	=BV	Rt Adj–Zero Fill	=RZ
Data Required	=DR	Self Check	=nxx
Dup Disable	=DD	nnCG (Check Gen)	
Field Exit Required	=FE	xx=Modulus	
Lower Case	=LC		

Editing — Functions

ADD (name)	PMT (prompt)
AUXDUP (name)	RANGE (low high)
AUXST (name)	RANGET (table [index],)
COMP (¹test fld1²@ . . . fldn [indicator])	RESET (=TOTn)
DSPATR (²attr . . .)	SEO (¹test)
EDTCDE (code [loat)	SETOF (ind)
ERROR (code [,'message'])	SETON (ind)
EXSR (subroutine)	SHIFT (⁴shift)
INSERT (fld1²@ . . . fldn)	SUB (name)
LOOK (table [index])	SUBST (table1 table2 [index])
	TADD (=TOTn)
¹test=EQ,GE,GT,LE,LT,NE	TSUB (=TOTn)
²attr=BL,CA,CS,HI,ND,RI,UL	XCHK (table index1 index2)
³@=+,-,/,	'literal'
⁴shift=A,D,H,N,V,W,X,Y	

1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20 21 22 23 24 25 26 27 28 29 30 31 32 33 34 35 36 37 38 39 40 41 42 43 44 45 46 47 48 49 50 51 52 53 54 55 56 57 58 59 60 61 62 63 64 65 66 67 68 69 70 71 72 73 74 75 76 77 78 79 80

*Number of sheets per pad may vary slightly.

The next two fields are fairly simple. The DESC field allows the operator to enter descriptive information about the item. The QUANT field requests information about the number of items that have been sold. The only unfamiliar entry for the QUANT field is the CHECK (RZ) keyword and parameter.

The CHECK(RZ) function places the operator's entry in the rightmost positions of the field and fills the unused leftmost positions with zeros. Remember that the DR parameter was used in the first job for the customer name field. It requires that data be entered in the field. The entry in the decimal usage column allows the field to be used later in calculations.

IBM International Business Machines Corporation

IBM 5280 DATA DESCRIPTION SPECIFICATIONS

Printed in U.S.A.

Job No. BILLING	Dataset	Keying Instruction	Graphic								Source Document DETAILED PURCHASE JOB	Page 3	of 8
Operator	Date		Key										

A

Sequence	Form Type	Comment (*)	Reserved	Indicator (for CHECK (BY, BV) or ERROR)	Reserved	Name Type (F/K/R/T)	Reserved	Dataset/Record/ Field/Table Name	Reserved	Length	Data Type	Reserved	Decimal Positions (0-9)	Usage (I/O/B/W)	Location: Screen Line	Location: Screen Pos	Editing: Checks=CHECK (code . . .)	Editing: Functions
1	A					R		DET										
2	A							DAT			6			I	00	20	01	AUXDUP(DATE)
3	A							SALS			3X			I	00	20	21	PMT(ENTER THE SALESMAN INITIALS)
4	A																	CHECK(DR BC)
5	A							A			1		0	W				
6	A							B			1		0	W				
7	A							ITEM			6C			I	00	20	31	PMT(ENTER THE ITEM CODE)
8	A																	SHIFT(DDXXXD)
9	A																	CHECK(DR)
10	A																	LOOK(ITEMT A)
11	A							DESC			30			I	00	30	01	PMT(ENTER THE DESCRIPTION)
12	A							QUANT			4D	0		I	00	30	41	PMT(ENTER THE QUANTITY)
13	A																	CHECK(RZ DR)
14	A																	
15	A																	
	A																	
	A																	
	A																	
	A																	

Checks=CHECK (code . . .)

Auto Dup	AD	Mandatory Entry	ME
Auto Skip	AS	Mandatory Fill	MF
Blank Check	BC	Rt Adj–Blank Fill	RB
Bypass	BY	Right to Left	RL
Bypass on Verify	BV	Rt Adj–Zero Fill	RZ
Data Required	DR	Self-Check	nnxx
Dup Disable	DD	n:C,G (Check Gen)	
Field Exit Required	FE	xx-Modulus	
Lower Case	LC		

Functions

ADD (name)
AUXDUP (name)
AUXST (name)
COMP (¹test fld1¹@ fldn [indicator ,])
DSPATR (²attr)
EDTCDE (code [float])
ERROR (code [,message])
EXSR (subroutine)
INSERT (fld¹@ fldn)
LOOK (table [index1])

¹test: EQ,GE,GT,LE,LT,NE
²attr: BL,CA,CS,HI,ND,RI,UL
³@ , *
⁴shift: A,D,H,N,V,W,X,Y

PMT (prompt)
RANGE (low,high)
RANGET (table [index1])
RESET (+TOTn)
SEQ (¹test)
SETOF (ind)
SETON (ind)
SHIFT (⁴shift)
SUB (name)
SUBST (table1 table2 [index1])
TADD (+TOTn)
TSUB (+TOTn)
XCHK (table index1 index2)
literal

1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20 21 22 23 24 25 26 27 28 29 30 31 32 33 34 35 36 37 38 39 40 41 42 43 44 45 46 47 48 49 50 51 52 53 54 55 56 57 58 59 60 61 62 63 64 65 66 67 68 69 70 71 72 73 74 75 76 77 78 79 80

*Number of sheets per pad may vary slightly.

The PRICE field is similar to the QUANT field. The 2 in the decimal usage column means that DE/RPG should consider the number that is entered to have two decimal positions. For example, if 245 were entered, DE/RPG should consider it to be 2.45 rather than 245.00.

This field also contains an index in the LOOK keyword parameter as did the ITEM field. These indexes will be used in this field to cross-check the validity of the item at a specified price using the INVENT table. Each index must be unique. For example, two A indexes cannot be specified in a program. If two A indexes were specified, the second value of A would replace the value of the first one, and the first value would be lost.

Notice how the indexes are used with the XCHK keyword and parameters. The first parameter is the name of the table (INVENT) to be checked and the second and third parameters are the indexes (A and B) provided earlier by the LOOK functions of the ITEM and PRICE fields.

IBM International Business Machines Corporation

IBM 5280 DATA DESCRIPTION SPECIFICATIONS

Printed in U.S.A.

Job No. BILLING	Dataset	Keying Instruction	Graphic / Key	Source Document DETAILED PURCHASE JOB	Page 3 of 8
Operator	Date				

A

Sequence	Form Type	Comment (*)	Reserved	Indicator (for CHECK (BY, BV) or ERROR)	Reserved	Name Type (F/K/R/T)	Reserved	Dataset/Record/ Field/Table Name	Reserved	Length	Data Type	Reserved	Decimal Positions (0-9)	Usage (I/O/B/W)	Location: Screen Line	Location: Screen Pos	Editing: Checks=CHECK (code.....) / Functions
1	A					R		DET									
2	A							DAT			6			I	00	2001	AUXDUP(DATE)
3	A							SALS			3X			I	00	2021	PMT(ENTER THE SALESMAN INITIALS)
4	A																CHECK(DR BC)
5	A							A			1		0	W			
6	A							B			1		0	W			
7	A							ITEM			6C			I	00	2031	PMT(ENTER THE ITEM CODE)
8	A																SHIFT(DDXXXD)
9	A																CHECK(DR)
10	A																LOOK(ITEMT A)
11	A							DESC			30			I	00	3001	PMT(ENTER THE DESCRIPTION)
12	A							QUANT			4D		0	I	00	3041	PMT(ENTER THE QUANTITY)
13	A																CHECK(RZ DR)
14	A							PRICE			5D		2	I	00	4001	PMT(ENTER THE PRICE)
15	A																LOOK(PRICET B)
	A																XCHK(INVENT A B)
	A																
	A																
	A																
	A																

Editing — Functions: ADD (name); AUXDUP (name); AUXST (name); COMP (test fld1[*@] fldn [indicator]); DSPATR (attr); EDTCDE (code [text]); ERROR (code [,message]); EXSR (subroutine); INSERT (fld1[*@] fldn); LOOK (table [,index]); PMT (prompt); RANGE (low,high); RANGET (table [,index]); RESET (*TOTn); SEQ (*test); SETOF (ind); SETON (ind); SHIFT (*shift); SUB (name); SUBST (table1,table2 [,index]); TADD (*TOTn); TSUB (*TOTn); XCHK (table,index1,index2); 'literal'

*test=EQ,GE,GT,LE,LT,NE; *attr=BL,CA,CS,HI,ND,RI,UL; *@=+,-,*,/; *shift=A,D,H,N,V,W,X,Y

*Number of sheets per pad may vary slightly.

The literal 'TOTAL:' and the COST field are specified next. You should understand literals and recognize that there must be an O (output) in the usage column. Now look at the COST field. Two new functions are shown here: (1) a calculation using named fields and the INSERT keyword and (2) a reserved word used for a counter.

To be able to use the named fields PRICE and QUANT in calculations be sure that the fields are named when they are described. If they are not named, they cannot be used. Also, look at the length of the COST field. You can determine this length by multiplying the largest numbers that can be provided for the price and quantity fields. Because you have a PRICE field that is 5 positions long and a QUANT field that is 4 positions long, the largest number you can have is 9999 x 999.99 which equals 9998900.01; therefore, a COST field with a length of 9 is sufficient. The decimal position is not counted.

Notice that the multiplication of the contents of the fields is designated by an *. Now, look at the TADD keyword and the *TOT1 parameter in this example. TADD causes this field to be added to the contents of the counter that is specified in the parameter. As directed by the supervisor, the counter you specified is counter 1 (*TOT1).

Using TADD and *TOT1 is a way to obtain an intermediate total. To understand this, consider how the job is used. Each purchase line of the sales order form equals one detail record. If you found the total of that line you would only have the total for a single purchase, not the total for the entire order. You need to add totals from all detail records for the same customer. The way you accomplish this is through the use of the counter. The counter will continue to add totals from the detail records until it is reset. All counters (*TOT) are 15 positions.

The DSPATR(HI) keyword and parameter highlight the contents of the field. Display attributes such as highlight require one display position before and after the field. The position column entries must reflect this requirement by leaving at least one blank before a field that uses the DSPATR keyword. The *POS location specifies the data location and not the attribute location. There is extra space between the literal TOTAL and the highlighted COST field to accommodate the display attribute position that is required.

IBM International Business Machines Corporation

IBM 5280 DATA DESCRIPTION SPECIFICATIONS

Printed in U.S.A.

Job No. BILLING	Dataset	Keying Instruction	Graphic	Source Document DETAILED PURCHASE JOB	Page 4 of 8
Operator	Date		Key		

A

Sequence	Form Type	Comment (*)	Reserved	Indicator (for CHECK (BY, BV) or ERROR)	Reserved	Name Type (F/K/R/T)	Reserved	Dataset/Record/Field/Table Name	Reserved	Length	Data Type	Decimal Positions (0-9)	Reserved	Usage (I/O/B/W)	Location: Line	Location: Pos	Editing
1	A														04	60	'TOTAL:'
2	A							COST		9		2		I	04	70	INSERT(PRICE*QUANT)
3	A																TADD(*TOT1)
4	A																DSPATR(HI)
5	A																
6	A																
7	A																
8	A																
9	A																
10	A																
11	A																
12	A																
13	A																
14	A																
15	A																
	A																
	A																
	A																
	A																
	A																

Checks=CHECK (code . . .)

Auto Dup	AD	Mandatory Entry	ME
Auto Skip	AS	Mandatory Fill	MF
Blank Check	BC	Rt Adj, Blank Fill	RB
Bypass	BY	Right to Left	RL
Right Adj, Zero Fill	RZ	Right Adj, Zero Fill	RZ
Data Required	DR	Self Check	nnxx
Dup Disable	DD	n: C,G (Check Gen)	
Field Exit Required	FE	xx: Modulus	
Lower Case	LC		

Functions

ADD (name)	PMT (prompt)
AUTODUP (name)	RANGE (low high)
AUXST (name)	RANGET (table [index])
COMP (test fldt fldb [M fldn indicator])	RESET (*TOTn)
DSPATR (attr [fldn])	SEQ (test)
EDTCDE (code float)	SETOF (ind)
ERROR (code [message])	SETON (ind)
EXSR (subroutine)	SHIFT (*shift)
INSERT (fldt fldb fldn)	SUB (name)
LOOK (table [index] test EQ,GE,GT,LE,LT,NE attr BL,CA,CS,HI,ND,RI,UL *shift A,D,H,N,V,W,X,Y	SUBST (table1 table2 [index])
	TADD (*TOTn)
	TSUB (*TOTn)
	XCHK (table index1 index2)
	literal

1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20 21 22 23 24 25 26 27 28 29 30 31 32 33 34 35 36 37 38 39 40 41 42 43 44 45 46 47 48 49 50 51 52 53 54 55 56 57 58 59 60 61 62 63 64 65 66 67 68 69 70 71 72 73 74 75 76 77 78 79 80

*Number of sheets per pad may vary slightly.

The customer number field is next.

IBM International Business Machines Corporation — IBM 5280 DATA DESCRIPTION SPECIFICATIONS — Printed in U.S.A.

Job No BILLING	Dataset	Keying Instructions	Graphic								
Operator	Date		Key								

Source Document DETAILED PURCHASE JOB	Page 4	of 8

A	Sequence	Form Type	Comment (*)	Reserved	Indicator (for CHECK (BY, BV) or ERROR)	Reserved	Name Type (F/K/R/T)	Reserved	Dataset/Record/ Field/Table Name	Reserved	Length	Data Type	Reserved	Decimal Positions (0-9)	Usage (I/O/B/W)	Location: Screen Line	Location: Screen Pos	Editing: Checks-CHECK (code ...)	Editing: Functions
	1 2 3 4 5	6	7	8 9 10	11 12 13 14 15 16		17	18	19 20 21 22 23 24 25 26	27 28 29	30 31 32 33 34	35	36	37	38	39 40 41	42 43 44	45 46 47 48 49 50 51 52 53 54 55 56 57 58 59 60 61 62 63 64 65 66 67 68 69 70 71 72 73 74 75 76 77 78 79 80	
	1	A														00	040	60'TOTAL:'	
	2	A							COST		9		2		I	00	407	0INSERT(PRICE*QUANT)	
	3	A																TADD(*TOT1)	
	4	A																DSPATR(HI)	
	5	A							CUSTNU		5				I	00	500	1PMT(ENTER THE CUSTOMER NUMBER)	
	6	A							MARK2		1				I	00	508	0INSERT('D')	
	7	A																	
	8	A																	
	9	A																	
	10	A																	
	11	A																	
	12	A																	
	13	A																	
	14	A																	
	15	A																	
		A																	
		A																	
		A																	
		A																	
		A																	

1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20 21 22 23 24 25 26 27 28 29 30 31 32 33 34 35 36 37 38 39 40 41 42 43 44 45 46 47 48 49 50 51 52 53 54 55 56 57 58 59 60 61 62 63 64 65 66 67 68 69 70 71 72 73 74 75 76 77 78 79 80

*Number of sheets per pad may vary slightly.

The next field is the record marker; D for detail record. The insert ('D') places the record mark in the display record.

If you have counted the field lengths, you know that the D mark will not automatically be in position 150 of the diskette record. In the first program, you positioned the record marker by using the CHECK(BY) function. In this program, you will reformat the record to position the marker. The description of the reformatted record occurs later in this chapter. Before you do that, however, you must complete the trailer display (TRAIL) record description.

You have completed the description of the detail display (DET) record. You are ready to describe the trailer display (TRAIL) record on the A specification.

Describing the TRAIL Record Using the A Specification

Most of the fields in the TRAIL record are automatic. The operator will have to enter only one field in the record. All functions used in this record description should by now be familiar to you. Briefly look at the contents as a review.

The DAY field allows the operator to enter the current date. The CUST field duplicates the contents of the named customer number field from the DET record. The TOT field places the total accumulated in the *TOT1 counter in the field and then resets the counter so that there is no carry-over balance from one customer to another.

The record marker T is the last field in this record. This marker will be positioned correctly by reformatting the record just as you will do for the DET record marker later in this chapter.

IBM International Business Machines Corporation — IBM 5280 DATA DESCRIPTION SPECIFICATIONS — Printed in U.S.A.

Job No. BILLING	Dataset	Keying Instruction	Graphic	Source Document DETAILED PURCHASE JOB	Page 5 of 8
Operator	Date		Key		

A

Sequence	Form Type	Comment (*)	Reserved	Indicator (for CHECK (BY, BV) or ERROR)	Reserved	Name Type (F/K/R/T)	Reserved	Dataset/Record/ Field/Table Name	Reserved	Length	Data Type	Reserved	Decimal Positions (0-9)	Usage (I/O/B/W)	Location: Screen Line	Location: Screen Pos	Editing
1	A					R		TRAIL									
2	A							DAY		6	D			I			PMT(ENTER THE CURRENT DATE)
3	A							CUST		5				I			INSERT(CUSTNU)
4	A							TOT		15			2	I			INSERT(*TOT1)
5	A																RESET(*TOT1)
6	A							MARK3		1				I			INSERT('T')
7	A																
8	A																
9	A																
10	A																
11	A																
12	A																
13	A																
14	A																
15	A																
	A																
	A																
	A																
	A																
	A																

Checks=CHECK (code . . .)

Auto Dup	-AD	Mandatory Entry	-ME
Auto Skip	-AS	Mandatory Fill	-MF
Blank Check	BC	Rt Adj–Blank Fill	-RB
Bypass	-BY	Right to Left	RL
Bypass on Verify	-BV	Rt Adj–Zero Fill	-RZ
Data Required	-DR	Self-Check	-mxx
Dup Disable	DD	xx=C,G,I(Check Gen)	
Field Exit Required	-FE	xx=Modulus	
Lower Case	LC		

Functions

ADD (name)	PMT (prompt)
AUXDUP (name)	RANGE (low high)
AUXST (name)	RANGET (table [index])
COMP (¹test fld1 ²@ fldn [indicator])	RESET (*TOTn)
DSPATR (²attr [ ])	SEQ (¹test)
EDTCDE (code [float])	SETOF (ind)
ERROR (code [message'])	SETON (ind)
EXSR (subroutine)	SHIFT (⁴shift)
INSERT (fld1 ³@ fldn)	SUB (name)
LOOK (table [index])	SUBST (table1 table2 [index])
	TADD (*TOTn)
¹test=EQ,GE,GT,LE,LT,NE	TSUB (*TOTn)
²attr=BL,CA,CS,HI,ND,RI,UL	XCHK (table index1 index2)
³@=+,-,*	'literal'
⁴shift: A,D,H,N,V,W,X,Y	

1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20 21 22 23 24 25 26 27 28 29 30 31 32 33 34 35 36 37 38 39 40 41 42 43 44 45 46 47 48 49 50 51 52 53 54 55 56 57 58 59 60 61 62 63 64 65 66 67 68 69 70 71 72 73 74 75 76 77 78 79 80

*Number of sheets per pad may vary slightly.

You have completed the definitions for the entry records used in the job. Before you continue, check your understanding of what you have learned:

- How many display records have been defined for this job?

  Answer: Three—SCRATCH, DET, and TRAIL

- Why have you defined a scratch record when the instructions do not describe one?

  Answer: To provide a date entry that can be used by the DET records and to instruct the operator in the use of the program.

- What does the INVENT table cross-check?

  Answer: That the entered item code matches the entered price

- What automatic calculation are you performing?

  Answer: Multiplying the QUANT and PRICE fields and maintaining an intermediate total for each customer.

If you understand the answers to these questions, continue reading. If you do not understand, reread the preceding sections on this topic.

DESCRIBING THE DATA SET FOR THIS PROGRAM

The description of the data set for this program will contain two records: the detail diskette (DETAIL) record and the trailer diskette (TRAILER) record. The description of the data set for the first program did not contain any record descriptions because no records in the first program were reformatted. The header record is not included in this description because the header record contents are being included in the data set by the manual copy function and, therefore, the record description does not have to be included in this diskette data set description. A file description line for the data set containing the header record will have to be included somewhere in the program, but you will see this later in this text.

When you reformat records in a data set, you must include a unique record name for each reformatted record (which, in this case is DETAIL and TRAILER). You must also list each named field in the record and provide a location for the field. All data fields in the reformatted record must be named. The location is the position you want the field to occupy in the diskette data set. You can rearrange fields in the data set, but you must use only fields from a single record for the same reformatted record and you must use all I (input) fields from the original record.

As you look at the sample that follows, notice that the first line in the data set description is a diskette file description statement with a name that matches the name in the TFILE parameter (BILLING). Next is a unique record name (DETAIL), followed by a list of named fields. The field names match those in the DET detail display record. Each field has an entry in the location columns on the A specifications.

The location entry is determined in the following way:

- The location of the first field is obviously 1 because it is first and, therefore, is in the first position of the record in the data set.

- The location of the second field is the position of the first field (1) plus the length of the first field (6) which equals 7.

- The locations of the remainder of the fields are calculated in this way until you come to the record marker. If you position the record marker field as you did the previous fields, it would appear in position 69 of the diskette data set. However, you want the marker in position 150 of the data set so that it matches the position of the header record marker. To do this, specify the position as 150.

When you reformat a record, you can specify any location you want for any field on the diskette data set. You can rearrange fields within the record, space them out or, as in this example, place one field in a predetermined position of the record.

Without looking at the reformatted TRAILER record, try to describe it on your A specification. The T mark should be in position 150 of the diskette data set. Remember, it must include all fields with I usage entries from the TRAIL record.

IBM International Business Machines Corporation

IBM 5280 DATA DESCRIPTION SPECIFICATIONS

Printed in U.S.A.

Job No. BILLING	Dataset	Keying Instruction	Graphic	Source Document DETAILED PURCHASE JOB	Page 6	of 8
Operator	Date		Key			

A

Sequence	Form Type	Comment (*)	Reserved	Indicator (for CHECK (BY, BV) or ERROR)	Reserved	Name Type (F/K/R/T)	Reserved	Dataset/Record/ Field/Table Name	Reserved	Length	Data Type	Reserved	Decimal Positions (0-9)	Usage (I/O/B/W)	Location: Screen Line	Location: Screen Pos	Editing
1 2 3 4 5	6	7	8	9 10	11 12 13 14 15 16	17	18	19 20 21 22 23 24 25 26	27 28 29	30 31 32 33 34	35	36	37	38	39 40 41	42 43 44	45 46 47 48 49 50 51 52 53 54 55 56 57 58 59 60 61 62 63 64 65 66 67 68 69 70 71 72 73 74 75 76 77 78 79 80
1	A					F		BILLING		150							DEVICE(DISK X'4000')
2	A					R		DETAIL									
3	A							DAT								1	
4	A							CUSTNU								7	
5	A							SALS								12	
6	A							ITEM								15	
7	A							DESC								21	
8	A							QUANT								51	
9	A							PRICE								55	
10	A							COST								60	
11	A							MARK2								150	
12	A					R		TRAILER									
13	A							DAY								1	
14	A							COSTNU								7	
15	A							TOT								12	
	A							MARK3								150	
	A																
	A																
	A																
	A																

Checks CHECK (code ...)

Alpha Only	AO	Mandatory Entry	ME
Alpha Shift	AS	Mandatory Fill	MF
Blank Check	BC	Right Adj, Blank Fill	RB
Bypass	BY	Right Adj, Zero Fill	RZ
Check Response	CR	Self Check	SC
Digits Only	DO	Signed Check (Check Digit)	SD
Field Exit Required	FE	Modulus	
Lowercase	LC		

Functions

ADD (name)	PMT (prompt)
AUXDUP (name)	RANGE (low high)
AUXST (name)	RANGET (table [index])
COMP (test hi/lo file ... hi/lo indicator)	RESET (+TOT+)
DSPATR (attr ...)	SEQ (test)
EDTCDE (code [mask])	SETOF (ind)
ERR/R (code [,message])	SETON (ind)
EXSR (subroutine)	SHIFT (*shift*)
INSERT (field file ... hi/lo)	SUB (name)
LOOK (table [,index ...])	SUBST (table1 table2 [,index])
	TADD (+TOT+)
test: EQ,GE,GT,LE,LT,NE	TSUB (+TOT+)
attr: BL,CA,CS,HI,ND,RI,UL	XCHK (table index1 index2)
literal	
*shift-A,D,H,N,V,W,X,Y	

1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20 21 22 23 24 25 26 27 28 29 30 31 32 33 34 35 36 37 38 39 40 41 42 43 44 45 46 47 48 49 50 51 52 53 54 55 56 57 58 59 60 61 62 63 64 65 66 67 68 69 70 71 72 73 74 75 76 77 78 79 80

*Number of sheets per pad may vary slightly.

You must provide three remaining data set descriptions: (1) a description for the data set containing the header record being copied into this data set and (2) descriptions for the table data sets used by the program.

DESCRIBING THE DATA SETS USED BY THE PROGRAM

You must include a file description line for the MASTHEAD data set from which you are copying header records.

The only information needed is the data set name and the DEVICE (DISK X'4000') keyword and parameter.

IBM International Business Machines Corporation

IBM 5280 DATA DESCRIPTION SPECIFICATIONS

Printed in U.S.A.

Job No. BILLING	Dataset	Keying Instruction	Graphic								Source Document DETAILED PURCHASE JOB	Page 7	of 8
Operator	Date		Key										

A	Sequence	Form Type	Comment (*)	Reserved	Indicator (for CHECK (BY, BV) or ERROR)	Reserved	Name Type (F/K/R/T)	Reserved	Dataset/Record/ Field/Table Name	Reserved	Length	Data Type	Reserved	Decimal Positions (0-9)	Usage (I/O/B/W)	Location: Screen: Line	Location: Screen: Pos	Editing: Checks=CHECK (code . . .)	Editing: Functions
1	10	A					F		MASTHEAD		150								DEVICE(DISK X'4000')
2	20	A																	
3	30	A																	
4	40	A																	
5	50	A																	
6	60	A																	
7	70	A																	
8	80	A																	
9	90	A																	
10	100	A																	
11	110	A																	
12	120	A																	
13	130	A																	
14	140	A																	
15	150	A																	
		A																	
		A																	
		A																	
		A																	
		A																	

Checks=CHECK (code . . .): Auto Dup AD; Auto Skip AS; Blank Check BC; Bypass BY; Bypass on Verify BV; Data Required DR; Dup Disable DD; Field Exit Required FE; Lower Case LC; Mandatory Entry ME; Mandatory Fill MF; Rt Adj–Blank Fill RB; Right to Left RL; Rt Adj–Zero Fill RZ; Self Check (nxx); n C G (Check Gen); xx–Modulus

Functions: ADD (name); AUXDUP (name); AUXST (name); COMP (1test fld1 2@ fldn [indicator]); DSPATR (2attr); EDTCDE (code float); ERROR (code ['message']); EXSR (subroutine); INSERT (fld1 fldn); LOOK (table [index]); PMT (prompt); RANGE (low high); RANGET (table [index]); RESET (+TOTn); SEQ (1test); SETOF (ind); SETON (ind); SHIFT (4shift); SUB (name); SUBST (table1 table2 [index]); TADD (+TOTn); TSUB (+TOTn); XCHK (table index1 index2); literal

1test: EQ,GE,GT,LE,LT,NE

2attr: BL,CA,CS,HI,ND,RI,UL

3@ + - *, /

4shift: A,D,H,N,V,W,X,Y

1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20 21 22 23 24 25 26 27 28 29 30 31 32 33 34 35 36 37 38 39 40 41 42 43 44 45 46 47 48 49 50 51 52 53 54 55 56 57 58 59 60 61 62 63 64 65 66 67 68 69 70 71 72 73 74 75 76 77 78 79 80

*Number of sheets per pad may vary slightly.

You must include a file description line for each table data set this program uses. Two table data sets are used:

- The TABLE1 data set, which contains the ITEMT and PRICET tables
- The TABLE2 data set, which contains the INVENT table

IBM International Business Machines Corporation — IBM 5280 DATA DESCRIPTION SPECIFICATIONS — Printed in U.S.A.

Job No. BILLING	Dataset	Keying Instruction	Graphic	Source Document DETAILED PURCHASE JOB	Page 8	of 8
Operator	Date		Key			

A

Sequence	Form Type	Comment (*)	Reserved	Indicator (for CHECK (BY, BV) or ERROR)	Reserved	Name Type (F,K,R,T)	Reserved	Dataset/Record/ Field/Table Name	Reserved	Length	Data Type	Reserved	Decimal Positions (0-9)	Usage (I/O/B/W)	Location: Screen: Line	Location: Screen: Pos	Editing: Checks: CHECK (code . . .)	Editing: Functions
10	A					F		TABLE1		11								DEVICE(DISK) X'4000')
20	A																	NUMENT(6)
30	A					T		ITEMT		6								
40	A					T		PRICET		5			2					
50	A					F		TABLE2		1								DEVICE(DISK) X'4000')
60	A																	NUMENT(12)
70	A					T		INVENT		1								
80	A																	
90	A																	
100	A																	
110	A																	
120	A																	
130	A																	
140	A																	
150	A																	
	A																	
	A																	
	A																	
	A																	
	A																	

Editing: Functions

ADD (name) | PMT (prompt)
AUXDUP (name) | RANGE (low:high)
AUXST (name) | RANGET (table (index))
COMP (*test fld1 *@ fld2 (indicator)) | RESET (+TOTn)
DSPATR (*attr *) | SEQ (*test)
EDTCDE (code float) | SETOF (ind)
ERROR (code | message) | SETON (ind)
EXSR (subroutine) | SHIFT (*shift)
INSERT (fld1 *@ fld2) | SUB (name)
LOOK (table (index)) | SUBST (table1 table2 (index))
TADD (+TOTn)
*test=EQ,GE,GT,LE,LT,NE | TSUB (+TOTn)
*attr=BL,CA,CS,HI,ND,RI,UL | XCHK (table index1 index2)
@=+,-,,/ | 'literal'
*shift=A,D,H,N,V,W,X,Y

1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20 21 22 23 24 25 26 27 28 29 30 31 32 33 34 35 36 37 38 39 40 41 42 43 44 45 46 47 48 49 50 51 52 53 54 55 56 57 58 59 60 61 62 63 64 65 66 67 68 69 70 71 72 73 74 75 76 77 78 79 80

*Number of sheets per pad may vary slightly.

Notice that the table data set descriptions are followed by lines with a letter T in column 17. This T specifies that the name in columns 19 through 24 identifies a table. For example, the data set named TABLE1 has two tables (ITEMT and PRICET). The TABLE2 data set has one table (INVENT). In the following chapter, you will learn how to create the tables. In this example, you are now learning only how to reference existing tables in a program that uses them. Each table description line (T in column 17) has a length entry. This entry tells DE/RPG the length of each entry, not the length of the table. The NUMENT keyword and numeric parameter tell DE/RPG the length of each table. When two tables are included in one data set, they must have the same number of entries (table length) but the length of the entries can be different.

You have now completed the diskette data set descriptions. You are ready to proceed to the job and format descriptions on the Z specification.

USING THE Z SPECIFICATION TO DESCRIBE THE JOB AND FORMATS

On the first line of the Z specification write the job description as defined by the instructions. It should look like this:

IBM International Business Machines Corporation

IBM 5280 GENERAL UTILITY SPECIFICATIONS

Printed in U. S. A.

Job DETAILED PURCHASE JOB		Keying Instruction	Graphic								Description CREATES DATA SET FOR MONTHLY BILLS	Page 1 of 8
Operator	Date		Key									

Z

Sequence	Form Type	Job/Format/Subroutine: Name Type	Format ID (1-9, A0-Z9)	Name	Reserved	Repeat (1-9, N)	Mode (E/R)	AND(A)	Test Conditions: Position to be Tested (*POSnnnn)	Reserved	Condition	Character to Test for ('C')	Reserved	Next Format ID (0-9, A0-Z9)	Reserved	Options: Job Line / Entry Lines
1	Z	J		TODAY					*POS		EQ					TFILE(BILLING)
2	Z								*POS		EQ					CFILE(MASTHEAD)
3	Z								*POS		EQ					
4	Z								*POS		EQ					
5	Z								*POS		EQ					
6	Z								*POS		EQ					
7	Z								*POS		EQ					
8	Z								*POS		EQ					
9	Z								*POS		EQ					
10	Z								*POS		EQ					
11	Z								*POS		EQ					
12	Z								*POS		EQ					
13	Z								*POS		EQ					
14	Z								*POS		EQ					
15	Z								*POS		EQ					
	Z								*POS		EQ					
	Z								*POS		EQ					
	Z								*POS		EQ					
	Z								*POS		EQ					
	Z								*POS		EQ					

Job Line: CFILE (data set) | DATE (*DMY/*YMD) | EDITC ([cuptd]) | ENTRATR (attr...) | EXITATR (attr...) | JOBOPT ([*NOPMT][*NOOPEN]) | PRTFILE (data set) | SHARE (names) | SHARER (names) | STATUS (name) | TFILE (data set [del/req]) | attr-BL CS HI ND RI (,)

Entry Lines: CLRL (number) | EOJ ([job dev [*PASS]]) | SLNO (line) | WRITE (name)

*Number of sheets per pad may vary slightly.

You should recognize the TFILE keyword and parameter. The CFILE keyword has not been explained. CFILE(MASTHEAD) means that you will be using the data set called MASTHEAD to copy data. This data set must be specified in a file description statement on the A specification (this has already been done). You must use this keyword and parameter in order for the copy keys to be valid. This completes the job description. Proceed to the format descriptions.

The entry format for the scratch record occurs first. Remember that the scratch display record contains the prompts telling the operator to use the copy function. The operator will need this information to know how to proceed. The name in columns 10 through 17 must match a record name on the A specification. The ID in columns 8 and 9 is arbitrary. If no ID has been specified by the supervisor, you may select any valid ID.

IBM International Business Machines Corporation

IBM 5280 GENERAL UTILITY SPECIFICATIONS

Printed in U. S. A.

Job DETAILED PURCHASE JOB | Keying Instruction | Graphic Key

Operator | Date

Description CREATES DATA SET FOR MONTHLY BILLS | Page 1 of 8

Z

Sequence	Form Type	Name Type	Format ID (1-9, A0-Z9)	Name	Reserved	Repeat (1-9, N)	Mode (E/R)	AND(A)	Position to be Tested (*POSnnnn)	Reserved	Condition	Character to Test for ('C')	Reserved	Next Format ID (0-9, A0-Z9)	Reserved	Options: Job Line / Entry Lines
1	Z	J		TODAY					*POS		EQ	' '				TFILE(BILLING)
2	Z								*POS		EQ	' '				CFILE(MASTHEAD)
3	Z		A0	SCRATCH		1	E		*POS		EQ	' '		X1		WRITE(*NO)
4	Z								*POS		EQ	' '				
5	Z								*POS		EQ	' '				
6	Z								*POS		EQ	' '				
7	Z								*POS		EQ	' '				
8	Z								*POS		EQ	' '				
9	Z								*POS		EQ	' '				
10	Z								*POS		EQ	' '				
11	Z								*POS		EQ	' '				
12	Z								*POS		EQ	' '				
13	Z								*POS		EQ	' '				
14	Z								*POS		EQ	' '				
15	Z								*POS		EQ	' '				
	Z								*POS		EQ	' '				
	Z								*POS		EQ	' '				
	Z								*POS		EQ	' '				
	Z								*POS		EQ	' '				
	Z								*POS		EQ	' '				

*Number of sheets per pad may vary slightly.

The 1 in column 20 tells DE/RPG to use the format once. The E indicates that the format is for the enter mode and the X1 in columns 45 and 46 specify the ID for the next format to be used for automatic format selection. The WRITE(*NO) keyword and parameter tell DE/RPG not to include the record in the diskette data set.

The first display record the operator will see is the scratch record. The operator will look at it and then use a field exit key to leave the display and proceed with the copy. The DET record will be displayed immediately following the scratch record. The scratch record will not be written in the diskette data set, but the header record will. Once the header record has been copied into the data set, the operator will see a display showing the format for the DET detail display record.

Because N appears in column 20, the DET format will be used until the operator selects the next format. The next format following the DET format is the TRAIL format (X2 in columns 45 and 46 of the DET format description). The WRITE(DETAIL) keyword and parameter for the DET format tells DE/RPG to use the DET format for the display but to write the data in the format specified by the DETAIL record description. (The DETAIL record was included in the diskette data set description.) This procedure shows you how to reformat records.

IBM International Business Machines Corporation

IBM 5280 GENERAL UTILITY SPECIFICATIONS

Printed in U. S. A.

Job DETAILED PURCHASE JOB | Keying Instruction | Graphic | Key

Operator | Date

Description CREATES DATA SET FOR MONTHLY BILLS | Page 1 of 8

Z

Sequence	Form Type	Name Type	Job/Format/Subroutine: Format ID (1-9, A0-Z9)	Job/Format/Subroutine: Name	Reserved	Repeat (1-9, N)	Mode (E/R)	AND(A)	Test Conditions: Position to be Tested (*POSnnnn)	Test Conditions: Reserved	Test Conditions: Condition	Test Conditions: Character to Test for ('C')	Reserved	Next Format ID (0-9, A0-Z9)	Reserved	Options: Job Line / Entry Lines
1	Z	U		TODAY					*POS		EQ	' '				DFILE(BILLING)
2	Z								*POS		EQ	' '				DFILE(MASTHEAD)
3	Z		A0	SCRATCH		1	E		*POS		EQ	' '		X1		WRITE(*NO)
4	Z		X1	DET		N	E		*POS		EQ	' '		X2		WRITE(DETAIL)
5	Z								*POS		EQ	' '				
6	Z								*POS		EQ	' '				
7	Z								*POS		EQ	' '				
8	Z								*POS		EQ	' '				
9	Z								*POS		EQ	' '				
10	Z								*POS		EQ	' '				
11	Z								*POS		EQ	' '				
12	Z								*POS		EQ	' '				
13	Z								*POS		EQ	' '				
14	Z								*POS		EQ	' '				
15	Z								*POS		EQ	' '				
	Z								*POS		EQ	' '				
	Z								*POS		EQ	' '				
	Z								*POS		EQ	' '				
	Z								*POS		EQ	' '				
	Z								*POS		EQ	' '				

Job Line: CFILE (data set); DATE (+DMY/+YMD); EDITC ([cuprtd]); ENTRATR (attr,...); EXITATR (attr,...); JOBOPT ([+NOPMT][+NOOPEN]); PRTFILE (data set); SHARE (names); SHARER (names); STATUS (name); TFILE (data set [delfreq]); attr=BL,CS,HI,ND,RI,UL

Entry Lines: CLRL (number); EOJ [(job dev [+PASS])]; SLNO (line); WRITE (name)

*Number of sheets per pad may vary slightly.

The TRAIL format (X2 in columns 8 and 9) is selected upon completion of the DET format. This record is also reformatted (WRITE(TRAILER)). Upon completion of the TRAIL format, which is used only once (1 in column 20), the scratch record is redisplayed.

IBM International Business Machines Corporation

IBM 5280 GENERAL UTILITY SPECIFICATIONS

Printed in U.S.A.

Job: DETAILED PURCHASE JOB | Keying Instruction | Graphic | Operator | Date | Key

Description: CREATES DATA SET FOR MONTHLY BILLS | Page 1 of 8

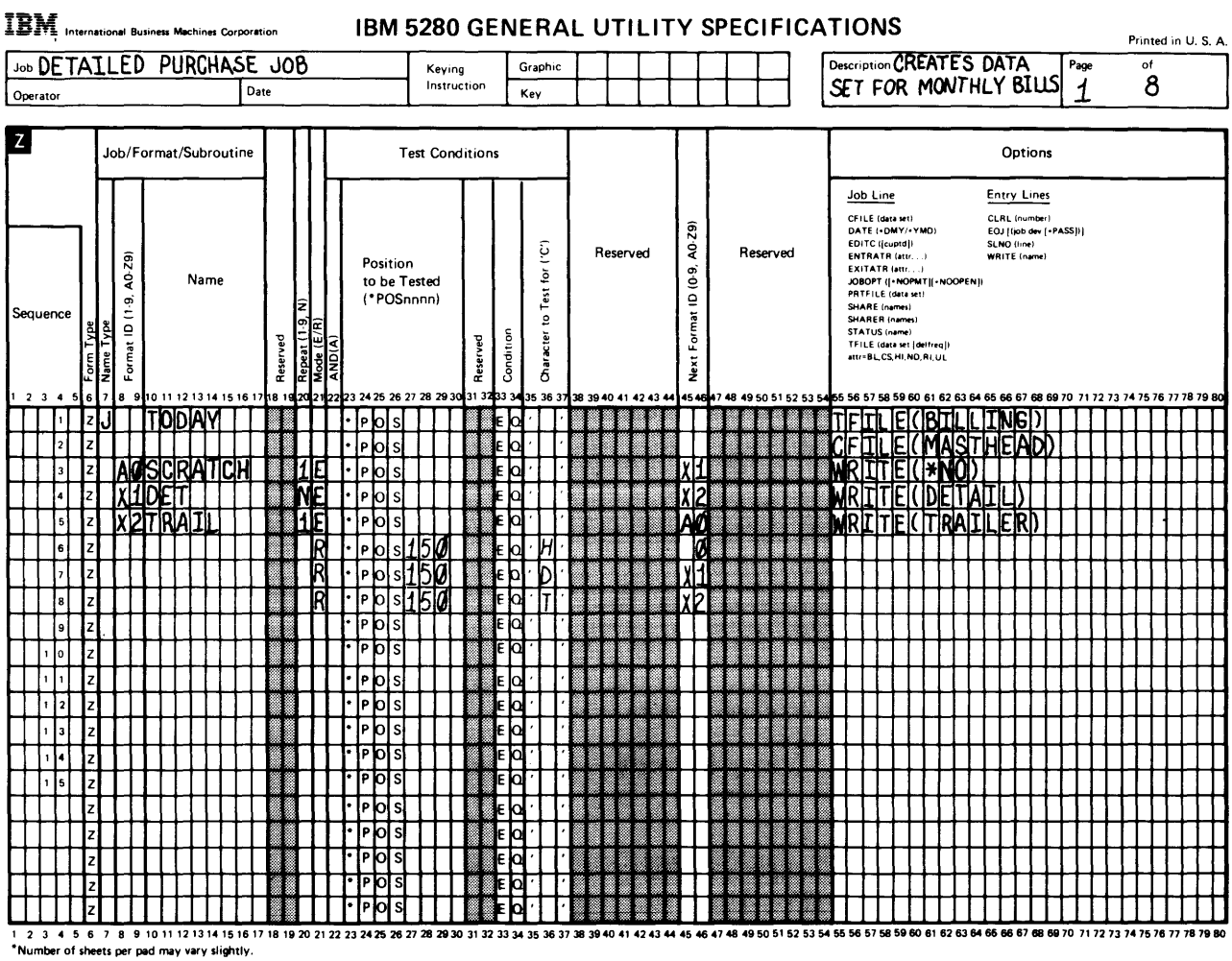
Z

Sequence	Form Type	Name Type	Job/Format/Subroutine: Format ID (1-9, A0-Z9)	Job/Format/Subroutine: Name	Reserved	Repeat (1-9, N)	Mode (E/R)	AND (A)	Test Conditions: Position to be Tested (*POSnnnn)	Reserved	Condition	Character to Test for ('C')	Reserved	Next Format ID (1-9, A0-Z9)	Reserved	Options: Job Line / Entry Lines
1	Z	J		TODAY					*POS		EQ					TFILE(BILLING)
2	Z								*POS		EQ					CFILE(MASTHEAD)
3	Z		A0	SCRATCH		1	E		*POS		EQ			X1		WRITE(*NO)
4	Z		X1	DET		N	E		*POS		EQ			X2		WRITE(DETAIL)
5	Z		X2	TRAIL		1	E		*POS		EQ			A0		WRITE(TRAILER)
6	Z								*POS		EQ					
7	Z								*POS		EQ					
8	Z								*POS		EQ					
9	Z								*POS		EQ					
10	Z								*POS		EQ					
11	Z								*POS		EQ					
12	Z								*POS		EQ					
13	Z								*POS		EQ					
14	Z								*POS		EQ					
15	Z								*POS		EQ					
	Z								*POS		EQ					
	Z								*POS		EQ					
	Z								*POS		EQ					
	Z								*POS		EQ					
	Z								*POS		EQ					

*Number of sheets per pad may vary slightly.

This process is continued for the enter mode (E in column 21) until the operator ends the job.

Once the enter mode description is complete, you must provide a description for the review modes. Place an R in column 21 of the Z specification. Write the reserved word *POS and the number 150, starting in column 24. Next, write 'H' in columns 35 through 37 and 0 in column 46; the apostrophes ('H') are required. This description tells DE/RPG to use format 0 (a default format that strings fields together across the display) for the display of a header record (an H in position 150) during the verification, update, and rerun modes. Next, write review descriptions for the detail and trailer records. The sample illustrates the coding.



*Number of sheets per pad may vary slightly.

Note: The scratch record can never be reviewed because it is not written to be displayed.

Whenever DE/RPG finds a D in position 150 of a diskette record, it should use format X1 (Det record) to display the data for the review mode. When it finds a T in position 150, it should use format X2 (TRAIL record) to display the data for the review mode. This allows you to use one format for the display and another for the diskette data set.

SUMMARY OF CHAPTER 8

You have completed the entire assignment you were initially given from the sales department. The only concept you need to learn is how to create table data sets. To teach you this, Chapter 9 creates data sets for the table used during the detail purchase job.

Before you proceed, try to answer the following questions:

1. Three tables are used in the program. What are their names?

   a. _____

   b. _____

   c. _____

2. Name three new functions you have learned in this chapter.

   a. _____

   b. _____

   c. _____

3. Answer T for true and F for false:

   a. Named fields can only be used with AUXST and AUXDUP. _____

   b. Fields do not have to be named. _____

   c. Indexes must be 3 positions long. _____

   d. All tables must have an index. _____

4. Mark the correct set of displays that will be seen when this program is used.

   a. Trailer, Detail, Header, Detail _____

   b. Detail (as needed), Trailer, Header _____

   c. Header, Detail (as needed), Trailer _____

   d. Scratch, Header, Detail (as needed), Trailer _____

5. Tell why you place the record identifier in the same position of each record.

   _____

   _____

Chapter 9. Creating Data Tables at Compile Time

In this chapter, you create three tables. The tables–ITEMT, PRICET, and INVENT–were used in the second sample job. There are two ways to create tables: (1) within the program that uses them and (2) within a separate data set. The first section in this chapter will show you how to create the ITEMT and PRICET tables within the second sample program. The second section will show you how to create the INVENT table in a new and separate diskette data set.

CREATING THE ITEMT AND PRICET TABLES WITHIN THE MASTER PROGRAM

When you defined the files and record descriptions for the tables that were used in the MASTER program, this is the way the A specification looked.

IBM International Business Machines Corporation — IBM 5280 DATA DESCRIPTION SPECIFICATIONS — Printed in U.S.A.

Job No. BILLING | Operator | Dataset | Date | Keying Instruction | Graphic | Key | Source Document DETAILED PURCHASE JOB | Page 8 of 8

A

Editing — Checks-CHECK (code . . .): Auto Dup AD; Auto Skip AS; Blank Check BC; Bypass BY; Bypass on Verify BV; Data Required DR; Dup Disable DD; Field Exit Required FE; Lower Case LC; Mandatory Entry ME; Mandatory Fill MF; Rt Adj–Blank Fill RB; Right to Left RL; Rt Adj–Zero Fill RZ; Self Check nxx; n C G (Check Gen); xx Modulus

Functions: ADD (name); AUXDUP (name); AUXST (name); COMP (¹test fld1²@ . . . fldn [indicator]); DSPATR (²attr . . .); EDTCDE (code float); ERROR (code ['message']); EXSR (subroutine); INSERT (fld1³@ . . . fldn); LOOK (table [index]); PMT (prompt); RANGE (low high); RANGET (table [index]); RESET (•TOTn); SEQ (¹test); SETOF (ind); SETON (ind); SHIFT (⁴shift); SUB (name); SUBST (table1 table2 (index)); TADD (•TOTn); TSUB (•TOTn); XCHK (table index1 index2); 'literal'

¹test EQ,GE,GT,LE,LT,NE
²attr BL,CA,CS,HI,ND,RI,UL
³@-+ - . •,
⁴shift=A,D,H,N,V,W,X,Y

Sequence (1-5)	Form Type (6)	Comment (*) (7)	Reserved (8)	Indicator (for CHECK (BY, BV) or ERROR) (9-10)	Reserved (11-16)	Name Type (F/K/R/T) (17)	Reserved (18)	Dataset/Record/Field/Table Name (19-26)	Reserved (27-29)	Length (30-34)	Data Type (35)	Reserved (36)	Decimal Positions (0-9) (37)	Usage (I/O/B/W) (38)	Location Screen Line (39-41)	Location Screen Pos (42-44)	Editing (45-80)
1	A					F		TABLE1		11							DEVICE(DISK X'4000')
2	A																NUMENT(6)
3	A					T		ITEMT		6							
4	A					T		PRICET		5			2				
5	A					F		TABLE2		1							DEVICE(DISK X'4000')
6	A																NUMENT(12)
7	A					T		INVENT		1							
8	A																
9	A																
10	A																
11	A																
12	A																
13	A																
14	A																
15	A																
	A																
	A																
	A																
	A																
	A																

*Number of sheets per pad may vary slightly.

Now you are going to change the table descriptions for the ITEMT and PRICET tables on the A specification so you can define the tables in the same program that uses them. These are called compile-time tables. To do this, you must change the table definitions somewhat and then you must add the table entries to the end of the program. The changes are shown in color in the following sample.

IBM International Business Machines Corporation — IBM 5280 DATA DESCRIPTION SPECIFICATIONS — Printed in U.S.A.

Job No BILLING	Dataset	Key ng Instruction	Graphic Key		Source Document DETAILED PURCHASE JOB	Page 8 of 8
Operator	Date					

A

Sequence	Form Type	Comment (*)	Reserved	Indicator (for CHECK (BY, BV) or ERROR)	Reserved	Name Type (F/K/R/T)	Reserved	Dataset/Record/Field/Table Name	Reserved	Length	Data Type	Reserved	Decimal Positions (0-9)	Usage (I/O/B/W)	Location: Line	Location: Pos	Editing
1	A					F		TABLE1		11							NUMENT(6)
2	A					T		ITEMT		6							
3	A					T		PRICET		5			2				
4	A					F		TABLE2		1							DEVICE(DISK X'4000')
5	A																NUMENT(12)
6	A					T		INVENT		1							
7	A																
	A																
	A																
	A																
	A																
	A																
	A																

**CTDATA TABLE1
00AAA10895
00AAA20015
00AAA30559
00AAA41847
00AAA50088
00AAA60349

*Number of sheets per pad may vary slightly.

The DEVICE keyword and parameter on the file description statement for the ITEMT and PRICET tables has been erased. Everything else within the body of the program remains as it had been for the original program. Now, look at the lines following the last description of the original program (the table description line for the INVENT table). Two asterisks mark the beginning of the table description. The name that follows the asterisks tells DE/RPG that this is a definition for a compile-time table. The table name follows and it matches a file description statement name for the tables that are being described within this program. The entries follow the asterisks and name. On each line, an entry for the ITEMT table is given and followed by an entry for the PRICET table. The value in the length column for the table description line in the program tells the program which entries belong to which table. The NUMENT keyword and parameter determine the length of the tables. Tables within the same data set must be the same length (although the length of the entries within the tables can differ). The *DE/RPG Reference Manual* contains detailed information about compile-time tables such as ITEMT and PRICET.

CREATING A SEPARATE DISKETTE DATA SET FOR THE INVENT TABLE

The original table description on the A specification for the BILLING program does not change when the INVENT table exists in a separate data set.

The program that creates the INVENT table is similar to the MASTER program.

IBM International Business Machines Corporation

IBM 5280 GENERAL UTILITY SPECIFICATIONS

Printed in U. S. A.

Job TAB JOB | Operator | Date | Keying Instruction | Graphic | Key

Description PROGRAM THAT CREATES THE INVENT TABLE | Page 1 of 2

Z

Sequence	Form Type	Job/Format/Subroutine: Name Type	Format ID (1-9, A0-Z9)	Name	Reserved	Repeat (1-9, N)	Mode (E/R)	AND(A)	Test Conditions: Position to be Tested (*POSnnnn)	Reserved	Condition	Character to Test for ('C')	Reserved	Next Format ID (0-9, A0-Z9)	Reserved	Options: Job Line / Entry Lines
1	Z	J		TABJOB					*POS		EQ	'				TFILE(TABLE2)
2	Z	A	1	INVENT		1	E		*POS		EQ	'		A2		
3	Z	A	2	INVENT2		1	E		*POS		EQ	'		A1		
4	Z								*POS		EQ	'				
5	Z								*POS		EQ	'				
6	Z								*POS		EQ	'				
7	Z								*POS		EQ	'				
8	Z								*POS		EQ	'				
9	Z								*POS		EQ	'				
10	Z								*POS		EQ	'				
11	Z								*POS		EQ	'				
12	Z								*POS		EQ	'				
13	Z								*POS		EQ	'				
14	Z								*POS		EQ	'				
15	Z								*POS		EQ	'				
	Z								*POS		EQ	'				
	Z								*POS		EQ	'				
	Z								*POS		EQ	'				
	Z								*POS		EQ	'				
	Z								*POS		EQ	'				

Job Line: CFILE (data set) | DATE (+DMY/+YMD) | EDITC ([cuptd]) | ENTRATR (attr ...) | EXITATR (attr ...) | JOBOPT ([+NOPMT][+NOOPEN]) | PRTFILE (data set) | SHARE (names) | SHARER (names) | STATUS (name) | TFILE (data set [delfreq]) | attr=BL,CS,HI,ND,RI,UL

Entry Lines: CLRL (number) | EOJ [(job dev [+PASS])] | SLNO (line) | WRITE (name)

*Number of sheets per pad may vary slightly.

IBM International Business Machines Corporation

IBM 5280 DATA DESCRIPTION SPECIFICATIONS

Printed in U.S.A.

Job No TABLE 2	Dataset	Keying Instruction	Graphic		Source Document PROGRAM THAT CREATES THE INVENT TABLE	Page 2	of 2
Operator	Date		Key				

A

Sequence	Form Type	Comment (*)	Reserved	Indicator (for CHECK (BY, BV) or ERROR)	Reserved	Name Type (F/K/R/T)	Reserved	Dataset/Record/ Field/Table Name	Reserved	Length	Data Type	Reserved	Decimal Positions (0-9)	Usage (I/O/B/W)	Location: Screen: Line	Location: Screen: Pos	Editing: Checks-CHECK (code ...) / Functions
1	A					F		FROM			1						DEVICE(CRT) DSPSIZ(16 80)
2	A					R		INVENT									PMT(ENTER THE NUMBER OF THE TABLE E+
3	A									1							NTRY FOR THE ITEM TABLE)
4	A																
5	A					R		INVENT2									
6	A									1							PMT(ENTER THE NUMBER OF THE TABLE E+
7	A																NTRY FOR THE PRICE TABLE
8	A					F		TABLE2		1							DEVICE(DISK X'4000')
9	A																
10	A																
11	A																
12	A																
13	A																
14	A																
15	A																
	A																
	A																
	A																
	A																
	A																

1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20 21 22 23 24 25 26 27 28 29 30 31 32 33 34 35 36 37 38 39 40 41 42 43 44 45 46 47 48 49 50 51 52 53 54 55 56 57 58 59 60 61 62 63 64 65 66 67 68 69 70 71 72 73 74 75 76 77 78 79 80

*Number of sheets per pad may vary slightly.

As you can see by looking at the sample, the Z specification must contain a job description statement and a format description statement for the record within the CRT file. Each table entry forms a record within the resulting diskette data set. The length of the record is specified in the file description statement for the diskette. The length column entry specifies the length of each table entry.

The advantages of creating tables in separate diskette data sets are that they can be used by more than one program, they can be created separately from the using program, and they can include prompts and literals that guide the operator in entering data for the tables.

SUMMARY OF CHAPTER 9

You have completed the chapter that tells you two ways to create tables. This concludes the teaching portion of the manual. Before you leave the chapter, try to answer the following questions.

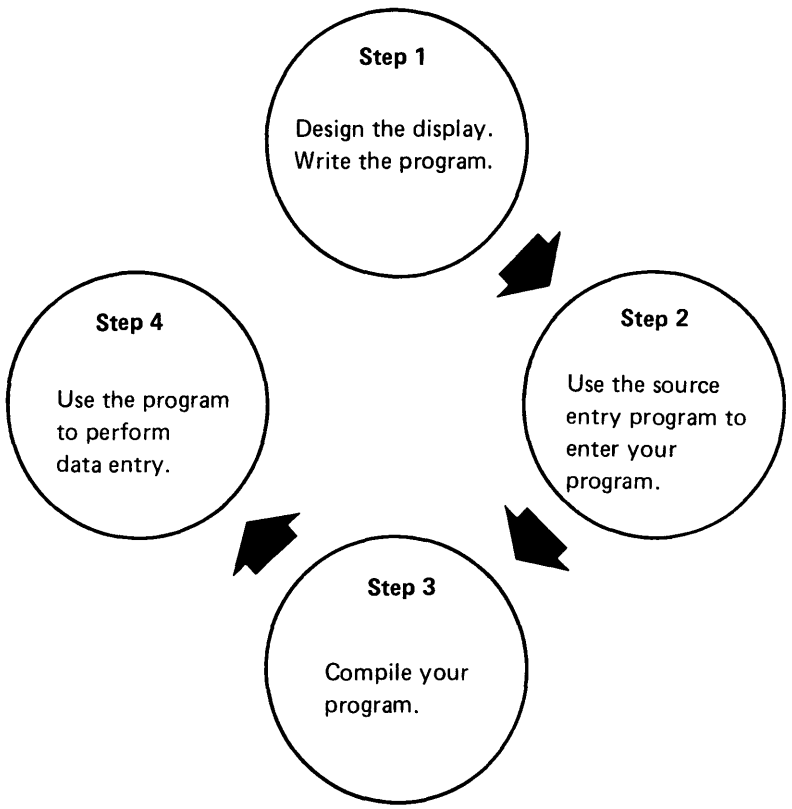
1. How many ways are available to create tables using DE/RPG? _____

   a. 3

   b. 1

   c. 2

2. Look at the following illustration. Write down the step number that reflects when compile-time tables are created. _____



3. Which of the following statements is true. _____

   When you create a compile-time table, you:

   a. Delete the NUMENT keyword and parameter.

   b. Replace the T in column 17 with an R.

   c. Delete the DEVICE keyword and parameter.

4. Which of the following statements is true:

   a. The INVENT table was created in a separate data set.

   b. The ITEMT and PRICET tables were created in separate data sets.

   c. The ITEMT, PRICET, and INVENT could be created in separate data sets or at compile time.

Compare your answers to those in Appendix A. If you feel you have sufficient understanding about creating tables, proceed to Chapter 10 and try to answer the final review questions for the manual. Chapter 10 contains questions about material in all the preceding chapters of the manual. Appendix A contains the answers. The reference after the question points to the chapter that contains the information.

Chapter 10. Self-Test Questions

Before you leave this manual, try to answer the questions in this chapter. The answers to the questions are in Appendix A. The references in italics after each question indicate the chapter in the text where the topic can be found.

1. Answer true or false to the following questions:

   a. Data entry is volume entry of data from a source document into a computing system. _____ *Chapter 2*

   b. A format is the smallest piece of data with which DE/RPG can work. _____ *Chapter 2*

   c. Edits and checks are special kinds of messages. _____ *Chapter 2*

   d. A file is related to an input/output device. _____ *Chapter 2*

2. The size of the display you have been using in the samples is _____ characters. *Chapter 3*

3. Row 1 of every display is reserved for the _____ _____. *Chapter 3*

4. _____ messages disappear when the associated field is exited but _____ remain on the display until the record is advanced. *Chapter 2 and 3*

5. All fields in the first program will be _____ to highlight their location to the operator. *Chapter 3*

6. Answer the following statements with T (true) or F (false).

   a. The device description statement for the diskette must always have an entry in the Length column of the A specification. _____ *Chapter 4*

   b. Field names must be no more than 8 characters long. _____ *Chapter 4*

   c. Job description statements are not necessary for all DE/RPG programs. _____ *Chapters 2 and 4*

   d. If there is only one record in the program, do not include an entry format description on the Z specification. _____ *Chapters 2 and 4.*

7. The name of the data set for the third sample program is _____. *Chapters 7 and 8.*

8. The _____ record is not written in the diskette data set for the second sample program. *Chapter 8.*

9. The _____ table determines if the business sells the item with the code that the operator entered. *Chapter 8.*

10. The _____ keyword and parameter provide data that can be duplicated or changed as needed. *Chapter 8.*

11. The CHECK(DR) keyword and parameter means _____. *Chapter 8.*

12. The indexes used in the second sample program are named _____ and _____. *Chapter 8.*

13. Tables can be created either at _____ _____ or in separate _____ _____. *Chapter 9.*

14. Because the ITEMT and PRICET tables are in the same data set, they must have the same number of _____. *Chapter 9.*

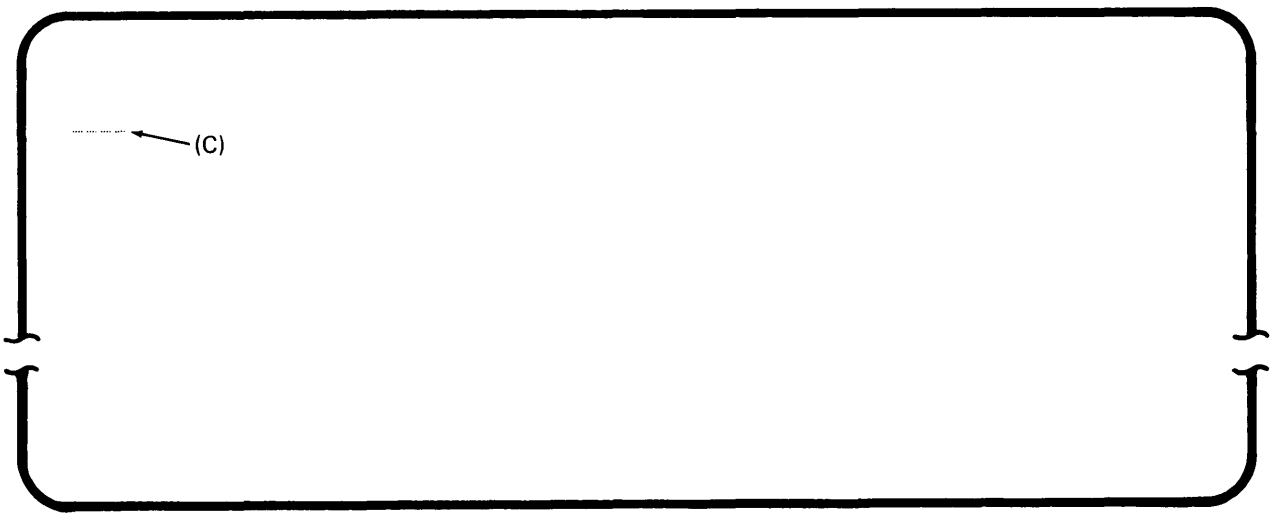
15. Try to complete the following program. The darkened areas on the specifications indicate information you should supply. Instructions are provided to aid you in this assignment.

Instructions

This program creates four records named ONE, TWO, THREE, and FOUR. The records are 9 positions long.

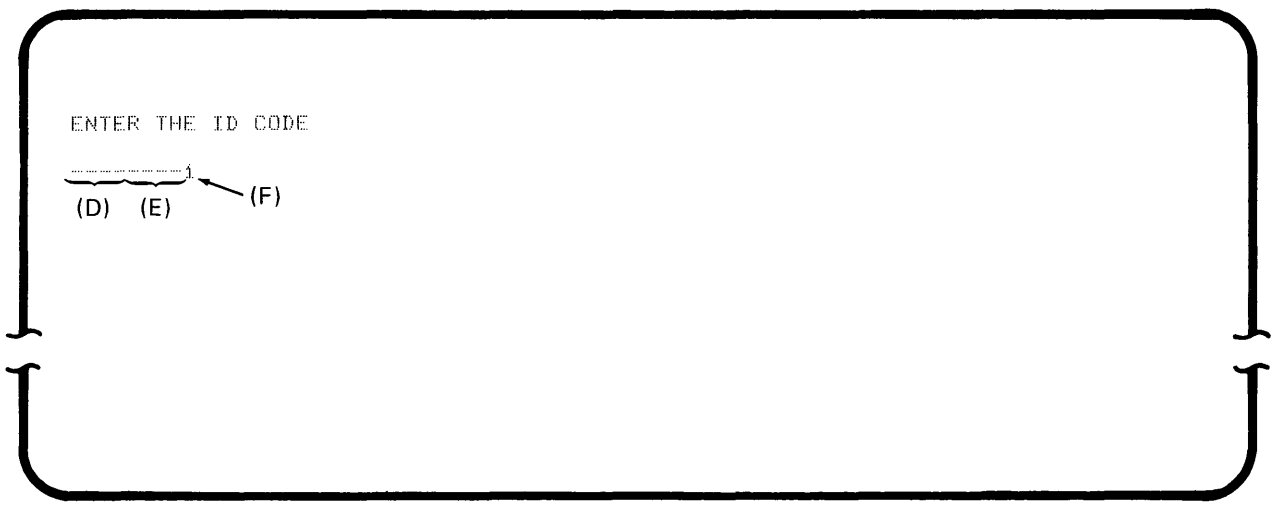
This is the appearance of the displays:

ONE

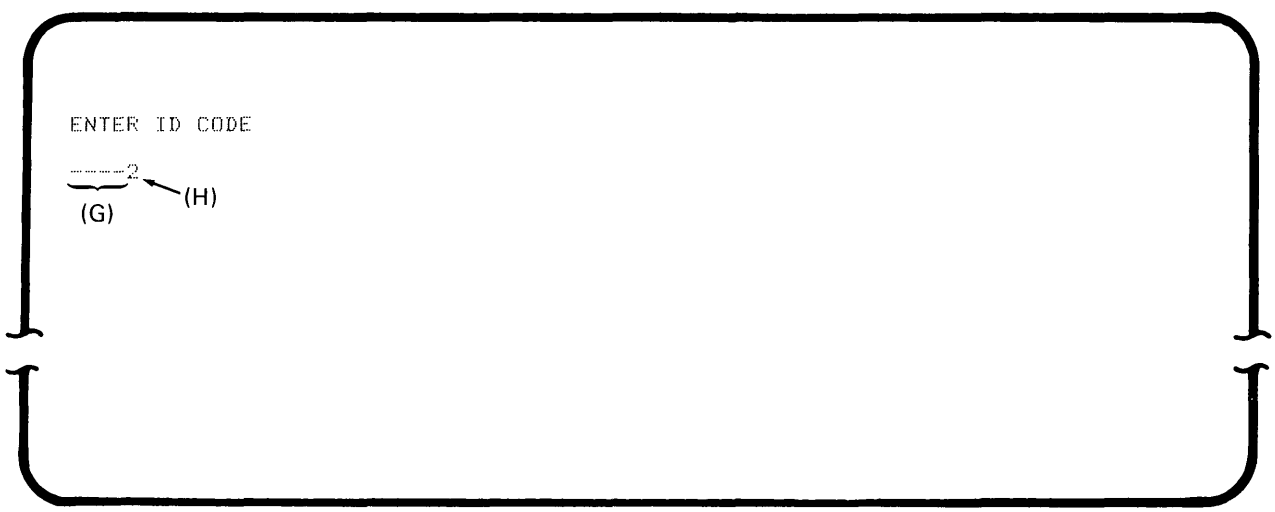


Note: A and B are not displayed.

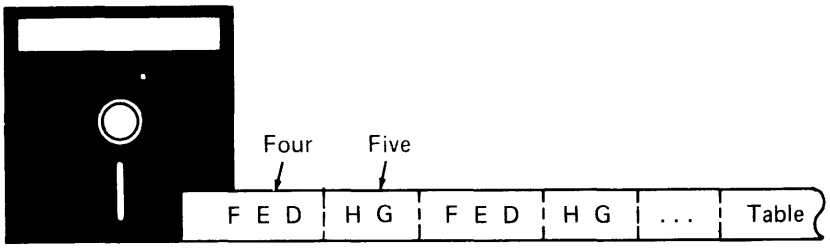
TWO



THREE



The resulting diskette data set looks like this:



IBM International Business Machines Corporation

IBM 5280 GENERAL UTILITY SPECIFICATIONS

Printed in U. S. A.

Job		Keying Instruction	Graphic	Description FINAL PROGRAM TEST	Page 1 of 3
Operator	Date		Key		

Z

Sequence	Form Type	Name Type	Format ID (1-9, A0-Z9)	Name	Reserved	Repeat (1-9, N)	Mode (E/R)	AND(A)	Position to be Tested (*POSnnnn)	Reserved	Condition	Character to Test for ('C')	Reserved	Next Format ID (0-9, A0-Z9)	Reserved	Options: Job Line	Options: Entry Lines
1	Z			TEST2					*POS		EQ	' '					(SAMPLE)
2	Z		W1	ONE		1	E		*POS		EQ	' '		W2			WRITE(*)
3	Z			TWO		1	E		*POS		EQ	' '		W3			WRITE(FOUR)
4	Z		W3	THREE					*POS		EQ	' '		W1			
5	Z								*POS	1	EQ	1 '					
6	Z						R		*POS		EQ	' '		W3			
7	Z								*POS		EQ	' '					
8	Z								*POS		EQ	' '					
9	Z								*POS		EQ	' '					
10	Z								*POS		EQ	' '					
11	Z								*POS		EQ	' '					
12	Z								*POS		EQ	' '					
13	Z								*POS		EQ	' '					
14	Z								*POS		EQ	' '					
15	Z								*POS		EQ	' '					
	Z								*POS		EQ	' '					
	Z								*POS		EQ	' '					
	Z								*POS		EQ	' '					
	Z								*POS		EQ	' '					
	Z								*POS		EQ	' '					

Job Line: CFILE (data set) | DATE (+DMY, +YMD) | EDITC ([cuptd]) | ENTRATR (attr ...) | EXITATR (attr ...) | JOBOPT ([+NOPMT][+NOOPEN]) | PRTFILE (data set) | SHARE (names) | SHARER (names) | STATUS (name) | TFILE (data set [,del,req]) | attr: BL,CS,HI,ND,RI,UL

Entry Lines: CLRL (number) | EOJ ([job dev [+PASS]]) | SLNO (line) | WRITE (name)

1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20 21 22 23 24 25 26 27 28 29 30 31 32 33 34 35 36 37 38 39 40 41 42 43 44 45 46 47 48 49 50 51 52 53 54 55 56 57 58 59 60 61 62 63 64 65 66 67 68 69 70 71 72 73 74 75 76 77 78 79 80

*Number of sheets per pad may vary slightly.

IBM International Business Machines Corporation

IBM 5280 DATA DESCRIPTION SPECIFICATIONS

Printed in U.S.A.

Job No	Dataset	Keying Instruction	Graphic								
Operator	Date		Key								

Source Document FINAL PROGRAM TEST	Page 2 of 3

A

Sequence | Form Type | Comment (*) | Reserved | Indicator (for CHECK (BY, BV) or ERROR) | Reserved | Name Type (F/K/R/T) | Reserved | Dataset/Record/Field/Table Name | Reserved | Length | Data Type | Reserved | Decimal Positions (0-9) | Usage (I/O/B/W) | Location: Line, Pos | Editing

Checks=CHECK (code . . .)

Auto Dup	AD	Mandatory Entry	ME
Auto Skip	AS	Mandatory Fill	MF
Blank Check	BC	Rt Adj–Blank Fill	RB
Bypass	BY	Right to Left	RL
Bypass on Verify	BV	Rt Adj–Zero Fill	RZ
Data Required	DR	Self Check	nnxx
Dup Disable	DD	n–C G (Check Gen)	
Field Exit Required	FE	xx–Modulus	
Lower Case	LC		

Functions

ADD (name)
AUXDUP (name)
AUXST (name)
COMP (*test fld1 *@ fldn [indicator])
DSPATR (*attr)
EDTCDE (code float)
ERROR (code ['message'])
EXSR (subroutine)
INSERT (fld1 *@ fldn)
LOOK (table [index])

*test–EQ,GE,GT,LE,LT,NE
*attr–BL,CA,CS,HI,ND,RI,UL
@–+,–,,/
*shift–A,D,H,N,V,W,X,Y

PMT (prompt)
RANGE (low high)
RANGET (table [index])
RESET (*TOTn)
SEQ (*test)
SETOF (ind)
SETON (ind)
SHIFT (*shift)
SUB (name)
SUBST (table1 table2 [index])
TADD (*TOTn)
TSUB (*TOTn)
XCHK (table index1 index2)
'literal'

Seq	Form Type	Name Type	Dataset/Record/Field/Table Name	Length	Data Type	Decimal Positions	Usage	Line	Pos	Editing
1	A	F		9						DEVICE(CRT)
2	A									
3	A	R	ONE							
4	A		A	2						
5	A		B	2						
6	A		C	4		2				INSERT(A*B)
7	A	R								
8	A		D	4			I			PMT(ENTER ID CODE)
9	A		E				I			INSERT(C)
10	A		F	1			I			INSERT('1')
11	A	R	THREE							
12	A		G	4						PMT(ENTER ID)
13	A									LOOK(IDTAB)
14	A		H							INSERT('2')
15	A									DEVICE(DISK X'4000')
	A	R	FOUR							
	A			1				1		
	A		E					2		
	A		D							
	A									

*Number of sheets per pad may vary slightly.

IBM International Business Machines Corporation

IBM 5280 DATA DESCRIPTION SPECIFICATIONS

Printed in U.S.A.

Job No	Dataset	Keying Instruction	Graphic							
Operator	Date		Key							

Source Document FINAL PROGRAM TEST | Page 3 of 3

A

Sequence	Form Type	Comment (*)	Reserved	Indicator (for CHECK (BY, BV) or ERROR)	Reserved	Name Type (F/K/R/T)	Reserved	Dataset/Record/ Field/Table Name	Reserved	Length	Data Type	Reserved	Decimal Positions (0-9)	Usage (I/O/B/W)	Location: Screen Line	Location: Screen Pos	Editing: Checks=CHECK (code...) / Functions
1 2 3 4 5 6	6	7	8	9 10	11 12 13 14 15 16	17	18	19 20 21 22 23 24 25 26	27 28 29	30 31 32 33 34	35	36	37	38	39 40 41	42 43 44	45 46 47 48 49 50 51 52 53 54 55 56 57 58 59 60 61 62 63 64 65 66 67 68 69 70 71 72 73 74 75 76 77 78 79 80
1	A					R		FIVE									
2	A							H									
3	A							G		4							
4	A					F		TABLE		4							NUMENT(3)
5	A							IDTAB					M				
6	A																
** 7	A																
1234 8	A																
5678 9	A																
9999 10	A																
11	A																
12	A																
13	A																
14	A																
15	A																
	A																
	A																
	A																
	A																
	A																

1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20 21 22 23 24 25 26 27 28 29 30 31 32 33 34 35 36 37 38 39 40 41 42 43 44 45 46 47 48 49 50 51 52 53 54 55 56 57 58 59 60 61 62 63 64 65 66 67 68 69 70 71 72 73 74 75 76 77 78 79 80

*Number of sheets per pad may vary slightly.

The answer for this program is in Appendix A.

Appendix A. Answers to the Test Questions at the End of Each Chapter

ANSWERS TO QUESTIONS IN CHAPTER 2

1.
   1. f
   2. c
   3. i
   4. j
   5. b
   6. d
   7. h
   8. e
   9. g
   10. a

2. Data entry is the process of transferring information from an existing source to a diskette record.

3. DE/RPG provides a way for you to write programs for data-entry jobs.

ANSWERS TO QUESTIONS IN CHAPTER 3

1.
   a. A customer address file
   b. 1. The data set consists of header records that contain information about each customer for the business

2. 1

3.

CORP (data)	CUSN (data)
STREET (data)	STATE (data)
CITY (data)	NUMBER (data)
H (data)	CUSTOMER NUMBER (literal)

ANSWERS TO QUESTIONS IN CHAPTER 4

1.
   a. Z
   b. A
   c. A
   d. Z and A
      The transaction file defines the data set on the Z specification. The file statement for the diskette defines the data set on the A specification.
   e. A
   f. Z and A
      The entry and review mode description on the Z specification and the record description on the A specification.
   g. A
   h. A

2.
   a. INPUT for the CRT file
   b. MASTHEAD for the diskette and transaction file

3. 007039
   Row 2 for the display is line 1 for the DE/RPG program.

4. c

5. a and c

6. review

ANSWERS TO SUMMARY QUESTIONS FOR CHAPTERS 2 THROUGH 6

1.
   a. T
      A job statement for a data-entry program must always contain the TFILE keyword and the data set name parameter.
   b. F
   c. T
      If no entry format is specified, no display appears.
   d. F
   e. F
   f. T
      Only one CRT file description is required for each job.
   g. F

h. T
At least one record must be defined in each program.

i. F

j. F
At least one field must be defined for each record. There is no maximum number of fields per record.

k. F
A prompt does not have to be included in a program.

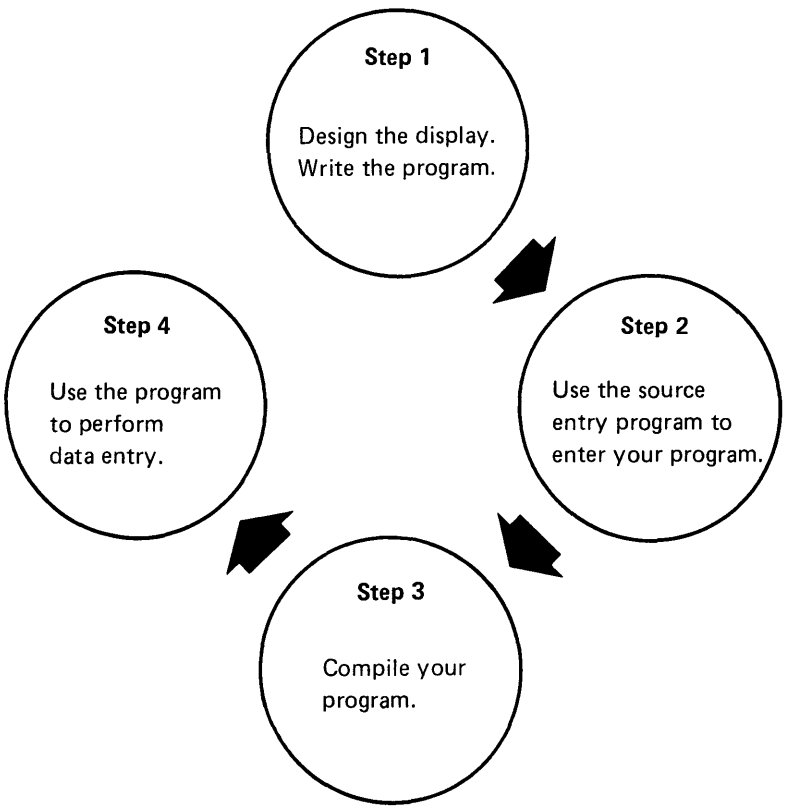
l. F
No edits or checks are required in a program.

m. T
A diskette file description must be included and its name must match that of the TFILE parameter.

n. F
Only one diskette file description is necessary for a simple data-entry program, although multiple file descriptions can be included in a single program.

2. b

3.



4. b and c
   a is false because the entry format name must match a record name on the A specification–not the CRT file name.

5. PMT(ENTER THE FOLLOWING INFORMATION)

6. b
   DE/RPG does provide a method for marking records that are in error. This is different from the record marker that allows DE/RPG to automatically select the appropriate format for entry and review modes.

7.

IBM International Business Machines Corporation — IBM 5280 DATA DESCRIPTION SPECIFICATIONS — Printed in U.S.A.

Job No.	Dataset	Keying Instruction	Graphic								Source Document TEST SAMPLE PROGRAM IN CHPT 6	Page of
Operator	Date		Key									

A

Sequence	Form Type	Comment (*)	Reserved	Indicator (for CHECK (BY, BV) or ERROR)	Reserved	Name Type (F/K/R/T)	Reserved	Dataset/Record/ Field/Table Name	Reserved	Length	Data Type	Reserved	Decimal Positions (0-9)	Usage (I/O/B/W)	Location: Line	Location: Pos	Editing
1	A					F		INPUT		30							DEVICE(CRT) DSPSIZ(6 80)
2	A					R		NAME									
3	A									30				I			PMT(ENTER YOUR NAME)
4	A																DSPATR(UL)
5	A																CHECK(DR)
6	A					R		ADDRESS									
7	A									30				I			PMT(ENTER YOUR ADDRESS)
8	A																DSPATR(UL)
9	A																CHECK(DR)
10	A					F		RESULT		30							DEVICE(DISK X'4000')
11	A																
12	A																
13	A																
14	A																
15	A																
	A																
	A																
	A																
	A																
	A																

Checks=CHECK (code . . .)

Auto Dup	:AD	Mandatory Entry	:ME
Auto Skip	:AS	Mandatory Fill	:MF
Blank Check	:BC	Rt Adj–Blank Fill	:RB
Bypass	:BY	Right to Left	:RL
Bypass on Verify	:BV	Rt Adj–Zero Fill	:RZ
Data Required	:DR	Self Check	:mxx
Dup Disable	:DD	n-C G (Check Gen)	
Field Exit Required	:FE	xx-Modulus	
Lower Case	:LC		

Functions

ADD (name)		PMT (prompt)
AUXDUP (name)		RANGE (low:high)
AUXST (name)		RANGE (table [:index])
COMP ('test:fld1'@	fldn [:indicator])	RESET (+TOTn)
DSPATR ('attr . . .')		SEQ ('test)
EDTCDE (code [float])		SETOF (ind)
ERROR (code ['message'])		SETON (ind)
EXSR (subroutine)		SHIFT ('shift)
INSERT (fld1'@	fldn)	SUB (name)
LOOK (table [:index:])		SUBST (table1:table2 [:index])
		TADD (+TOTn)
'test=EQ,GE,GT,LE,LT,NE		TSUB (+TOTn)
'attr=BL,CA,CS,HI,ND,RI,UL		XCHK (table:index1:index2)
'@=+ – *,/		'literal'
*shift A,D,H,N,V,W,X,Y		

1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20 21 22 23 24 25 26 27 28 29 30 31 32 33 34 35 36 37 38 39 40 41 42 43 44 45 46 47 48 49 50 51 52 53 54 55 56 57 58 59 60 61 62 63 64 65 66 67 68 69 70 71 72 73 74 75 76 77 78 79 80

*Number of sheets per pad may vary slightly.

IBM International Business Machines Corporation

IBM 5280 GENERAL UTILITY SPECIFICATIONS

Printed in U. S. A.

Job		Keying Instruction	Graphic							
Operator	Date		Key							

Description	Page of

Z

Sequence	Form Type	Job/Format/Subroutine: Name Type	Format ID (1-9, A0-Z9)	Name	Reserved	Repeat (1-9, N)	Mode (E/R)	Test Conditions: AND/(A)	Position to be Tested (*POSnnnn)	Reserved	Condition	Character to Test for ('C')	Reserved	Next Format ID (0-9, A0-Z9)	Reserved	Options
1 2 3 4 5	6	7	8 9	10 11 12 13 14 15 16 17	18 19	20	21	22	23 24 25 26 27 28 29 30	31 32	33 34	35 36 37	38 39 40 41 42 43 44	45 46	47 48 49 50 51 52 53 54	55 56 57 58 59 60 61 62 63 64 65 66 67 68 69 70 71 72 73 74 75 76 77 78 79 80
1	Z	J		EXAMPLE					*POS		EQ	' '				TFILE(RESULT 10)
2	Z	B	1	NAME		1	E		*POS		EQ	' '		B	2	
3	Z	B	2	ADDRES		1	E		*POS		EQ	' '		B	1	
4	Z					R			*POS		EQ	' '		B	1	
5	Z								*POS		EQ	' '				
6	Z								*POS		EQ	' '				
7	Z								*POS		EQ	' '				
8	Z								*POS		EQ	' '				
9	Z								*POS		EQ	' '				
10	Z								*POS		EQ	' '				
11	Z								*POS		EQ	' '				
12	Z								*POS		EQ	' '				
13	Z								*POS		EQ	' '				
14	Z								*POS		EQ	' '				
15	Z								*POS		EQ	' '				
	Z								*POS		EQ	' '				
	Z								*POS		EQ	' '				
	Z								*POS		EQ	' '				
	Z								*POS		EQ	' '				
	Z								*POS		EQ	' '				

Job Line: CFILE (data set); DATE (+DMY/+YMD); EDITC ([cuptd]); ENTRATR (attr...); EXITATR (attr...); JOBOPT ([+NOPMT][+NOOPEN]); PRTFILE (data set); SHARE (names); SHARER (names); STATUS (name); TFILE (data set [delfreq]); attr=BL,CS,HI,ND,RI,UL

Entry Lines: CLRL (number); EOJ [(job dev [+PASS])]; SLNO (line); WRITE (name)

1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20 21 22 23 24 25 26 27 28 29 30 31 32 33 34 35 36 37 38 39 40 41 42 43 44 45 46 47 48 49 50 51 52 53 54 55 56 57 58 59 60 61 62 63 64 65 66 67 68 69 70 71 72 73 74 75 76 77 78 79 80

*Number of sheets per pad may vary slightly.

ANSWERS TO QUESTIONS IN CHAPTER 7

1. a, c, and d
   The record types are (1) header, (2) detail, and (3) trailer.
   The detail and trailer records are reformated.
   The three tables that are used are ITEMT, PRICET, and INVENT.
   Figure 7 illustrates the source document for the second sample job.

2. a and c
   Fields are normally named so they can be used more than once within the same program.
   A table is a list of items.
   A grocery list is an example of a table.

ANSWERS TO QUESTIONS IN CHAPTER 8

1.
   a. ITEMT
   b. PRICET
   c. INVENT

2.
   a. Using tables
   b. Performing calculations
   c. Reformatting fields in diskette records
   d. Specifying formats for automatic selection in the review mode
   e. Using AUXST and AUXDUP
   f. Using indexes
   g. Suppressing the writing of a record
   h. Using a scratch record
   i. Using multiple records
   j. Copying records from another data set
   k. Specifying data sets for sharing

3.
   a. F
      Named fields can be used in a variety of ways. If a field is used with AUXDUP or AUXST, it must be named.
   b. T
      Only fields that will be reused must be named.
   c. F
      Indexes must be as long as necessary to hold the maximum number of positions in the associated table.
   d. F
      Indexes must only be used when the associated tables are used in edits that require them such as XCHK.

4. d

5. To select the appropriate review format

ANSWERS TO QUESTIONS IN CHAPTER 9

1. c
   (1) in the same program in which they are used and (2) in a separate data set.

2. 2

3. c

4. a and c
   B is incorrect because the ITEMT and PRICET tables were created within the using program.

ANSWERS TO QUESTIONS IN CHAPTER 10

1.
   a. T
   b. F
      A field is the smallest piece.
   c. F
      Prompts and literals are special messages.
   d. T

2. 480

3. status line

4. prompt literals

5. underlined

6.
   a. T
      If there is no length, the program cannot be compiled.
   b. F
      The maximum field length is 6.
   c. F
      Each program must have a job statement.
   d. F
      Each program must have an entry format description.

7. BILLING

8. SCRATCH

9. ITEMT

10. AUXST

11. nonblank data required

12. A B

13. compile time data sets

14. entries

IBM International Business Machines Corporation

IBM 5280 GENERAL UTILITY SPECIFICATIONS

Printed in U. S. A.

Job		Keying Instruction	Graphic	
Operator	Date		Key	

Description: FINAL PROGRAM TEST ANSWERS — Page 1 of 3

Z

Sequence	Form Type	Job/Format/Subroutine: Name Type	Format ID (1-9, A0-Z9)	Name	Reserved	Repeat (1-9, N)	Mode (E/R)	AND/(A)	Test Conditions: Position to be Tested (*POSnnnn)	Reserved	Condition	Character to Test for ('C')	Reserved	Next Format ID (0-9, A0-Z9)	Reserved	Options: Job Line / Entry Lines
1	Z	J		TEST2					*POS		EQ	' '				TFILE(SAMPLE)
2	Z		W1	ONE		1	E		*POS		EQ	' '		W2		WRITE(*NO)
3	Z		W2	TWO		1	E		*POS		EQ	' '		W3		WRITE(FOUR)
4	Z		W3	THREE		1	E		*POS		EQ	' '		W1		WRITE(FIVE)
5	Z						R		*POS1		EQ	'1'		W2		
6	Z						R		*POS1		EQ	'2'		W3		
7	Z								*POS		EQ	' '				
8	Z								*POS		EQ	' '				
9	Z								*POS		EQ	' '				
10	Z								*POS		EQ	' '				
11	Z								*POS		EQ	' '				
12	Z								*POS		EQ	' '				
13	Z								*POS		EQ	' '				
14	Z								*POS		EQ	' '				
15	Z								*POS		EQ	' '				
	Z								*POS		EQ	' '				
	Z								*POS		EQ	' '				
	Z								*POS		EQ	' '				
	Z								*POS		EQ	' '				
	Z								*POS		EQ	' '				

1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20 21 22 23 24 25 26 27 28 29 30 31 32 33 34 35 36 37 38 39 40 41 42 43 44 45 46 47 48 49 50 51 52 53 54 55 56 57 58 59 60 61 62 63 64 65 66 67 68 69 70 71 72 73 74 75 76 77 78 79 80

*Number of sheets per pad may vary slightly.

IBM International Business Machines Corporation

IBM 5280 DATA DESCRIPTION SPECIFICATIONS

Printed in U.S.A.

Job No.	Dataset	Keying Instruction	Graphic							
Operator	Date		Key							

Source Document: FINAL PROGRAM TEST ANSWERS — Page 2 of 3

A

Sequence	Form Type	Comment (*)	Reserved	Indicator (for CHECK (BY, BV) or ERROR)	Reserved	Name Type (F/K/R/T)	Reserved	Dataset/Record/ Field/Table Name	Reserved	Length	Data Type	Reserved	Decimal Positions (0-9)	Usage (I/O/B/W)	Location: Screen Line	Location: Screen Pos	Editing
1	A					F		SCREEN		9							DEVICE(CRT)
2	A																DSPSIZ(6 80)
3	A					R		ONE									
4	A							A		2				2W			
5	A							B		2				2W			
6	A							C		4				2I			INSERT(A*B)
7	A					R		TWO									
8	A							D		4				I			PMT(ENTER ID CODE)
9	A							E		4				2I			INSERT(C)
10	A							F		1				I			INSERT('I')
11	A					R		THREE									
12	A							G		4				0I			PMT(ENTER ID CODE)
13	A																LOOK(IDTAB)
14	A							H		1				I			INSERT('2')
15	A					F		SAMPLE		9							DEVICE(DISK X'4000')
	A					R		FOUR									
	A							F		1						1	
	A							E		4			2			2	
	A							D		4						6	
	A																

Location

Editing

Checks=CHECK (code . . .)

Auto Dup	=AD	Mandatory Entry	=ME
Auto Skip	=AS	Mandatory Fill	=MF
Blank Check	=BC	Rt Adj–Blank Fill	=RB
Bypass	=BY	Right to Left	=RL
Bypass on Verify	=BV	Rt Adj–Zero Fill	=RZ
Data Required	=DR	Self-Check	=nxx
Dup Disable	=DD	n=C,G (Check Gen)	
Field Exit Required	=FE	xx=Modulus	
Lower Case	=LC		

Functions

ADD (name)
AUXDUP (name)
AUXST (name)
COMP (¹test fld1²@ . . . fldn [indicator,])
DSPATR (²attr . . .)
EDTCDE (code [float])
ERROR (code [³message])
EXSR (subroutine)
INSERT (fld1²@ . . . fldn)
LOOK (table [index])
PMT (prompt)
RANGE (low high)
RANGET (table [index])
RESET (+TOTn)
SEQ (¹test)
SETOF (ind)
SETON (ind)
SHIFT (⁴shift)
SUB (name)
SUBST (table1 table2 [index])
TADD (+TOTn)
TSUB (+TOTn)
XCHK (table index1 index2)

¹test=EQ,GE,GT,LE,LT,NE
²attr=BL,CA,CS,HI,ND,RI,UL
³@=+,–,*,/
⁴shift=A,D,H,N,V,W,X,Y
'literal'

*Number of sheets per pad may vary slightly.

IBM International Business Machines Corporation

IBM 5280 DATA DESCRIPTION SPECIFICATIONS

Printed in U.S.A.

Job No	Dataset	Keying Instruction	Graphic	Source Document FINAL PROGRAM TEST ANSWERS	Page 3	of 3
Operator	Date		Key			

A

Sequence (1–5)	Form Type (6)	Comment (*) (7)	Name Type (F/K/R/T) (17)	Dataset/Record/Field/Table Name (19–26)	Length (30–34)	Data Type (35)	Decimal Positions (0-9) (37)	Usage (I/O/B/W) (38)	Location Line (39–41)	Location Pos (42–44)	Editing (45–80)
1	A		R	FIVE							
2	A			H	1					1	
3	A			G	4		Ø			2	
4	A		F	TABLE	4						NUMENT(3)
5	A		T	IDTAB	4		Ø				
6	A										
**CTDATA TABLE											
12348	A										
56789	A										
99990	A										
11	A										
12	A										
13	A										
14	A										
15	A										
	A										
	A										
	A										
	A										
	A										

Editing — Checks=CHECK (code . . .): Auto Dup AD, Auto Skip AS, Blank Check BC, Bypass BY, Bypass on Verify BV, Data Required DR, Dup Disable DD, Field Exit Required FE, Lower Case LC, Mandatory Entry ME, Mandatory Fill MF, Right Adjust Blank Fill RB, Right to Left RL, Right Adjust Zero Fill RZ.

Functions: ADD (name), AUXDUP (name), AUXST (name), COMP, DSPATR (attr), EDTCDE (code float), ERROR (code message), EXSR (subroutine), INSERT, LOOK (table index), PMT (prompt), RANGE (low high), RANGET (table index), RESET (*TOTn), SEQ (test), SETOF (ind), SETON (ind), SHIFT (shift), SUB (name), SUBST (table1 table2 index), TADD (*TOTn), TSUB (*TOTn), XCHK (table index1 index2).

test: EQ,GE,GT,LE,LT,NE; attr: BL,CA,CS,HI,ND,RI,UL; shift: A,D,H,N,V,W,X,Y

*Number of sheets per pad may vary slightly

Appendix B. Glossary

alphabetic fields: Fields that accept character rather than numeric entries.

auto dup: Refers to a switched key (or the function caused by using the switch) which enables the automatic duplication of fields specified with the AUXDUP keyword.

automatic functions: Actions performed by the DE/RPG program without requiring operator intervention. An example is the automatic insertion of a record marker as specified.

checking: The automatic verification of the correctness of the type of entry such as requiring all nonblank characters to fill a field.

copying: The duplication of a set of information.

counters: Fifteen-byte areas of storage that are represented by the names *TOT1 through *TOT9 and that can be used to contain intermediate results.

data entry: The transfer of data from a source document to a diskette data set.

data required fields: A field edit that requires the operator to enter nonblank data in the field before leaving it.

data set: A collection of related data records on a diskette.

diskette: The media used to record data.

display: The data that is shown on the screens of the data stations.

display work sheets: A design tool which enables you to lay out the contents of the displays you want to create with the program.

EOJ function: The use of the CMD key followed by the numeric 7 key (typewriter keyboard) for the purpose of terminating the job.

editing: The manipulation of data in a field to enhance its usability such as adjusting the entry to the right of the field and filling the remaining positions with zeros or blanks.

fields: Small pieces of related data, which together make up records; either consisting of data entered by an operator or of automatically supplied data.

files: A temporary storage area for data before it is displayed or written on a diskette.

formats: The definition of the sequence of data for the displays and diskettes.

job: The definition and control of a data-entry task.

line: The relative horizontal location across the display. See also row.

literals: Messages displayed as located by the program.

modes: Types of operations during which data entry using DE/RPG can be performed. The four modes are enter, update, verify, and rerun.

program: The information contained on the Z and A specification to describe the data entry job.

prompts: Messages displayed on the second line of the display.

record advance: Refers to a key that, when pressed, displays the format for the next record.

records: A unit of related fields, equal size units of which make up the diskette data set.

reformatting: The rearranging of data into a sequence that differs from that used for the enter mode.

row: The physical horizontal location across the display. See also line.

source document: The object from which the operator is taking data to enter into the system.

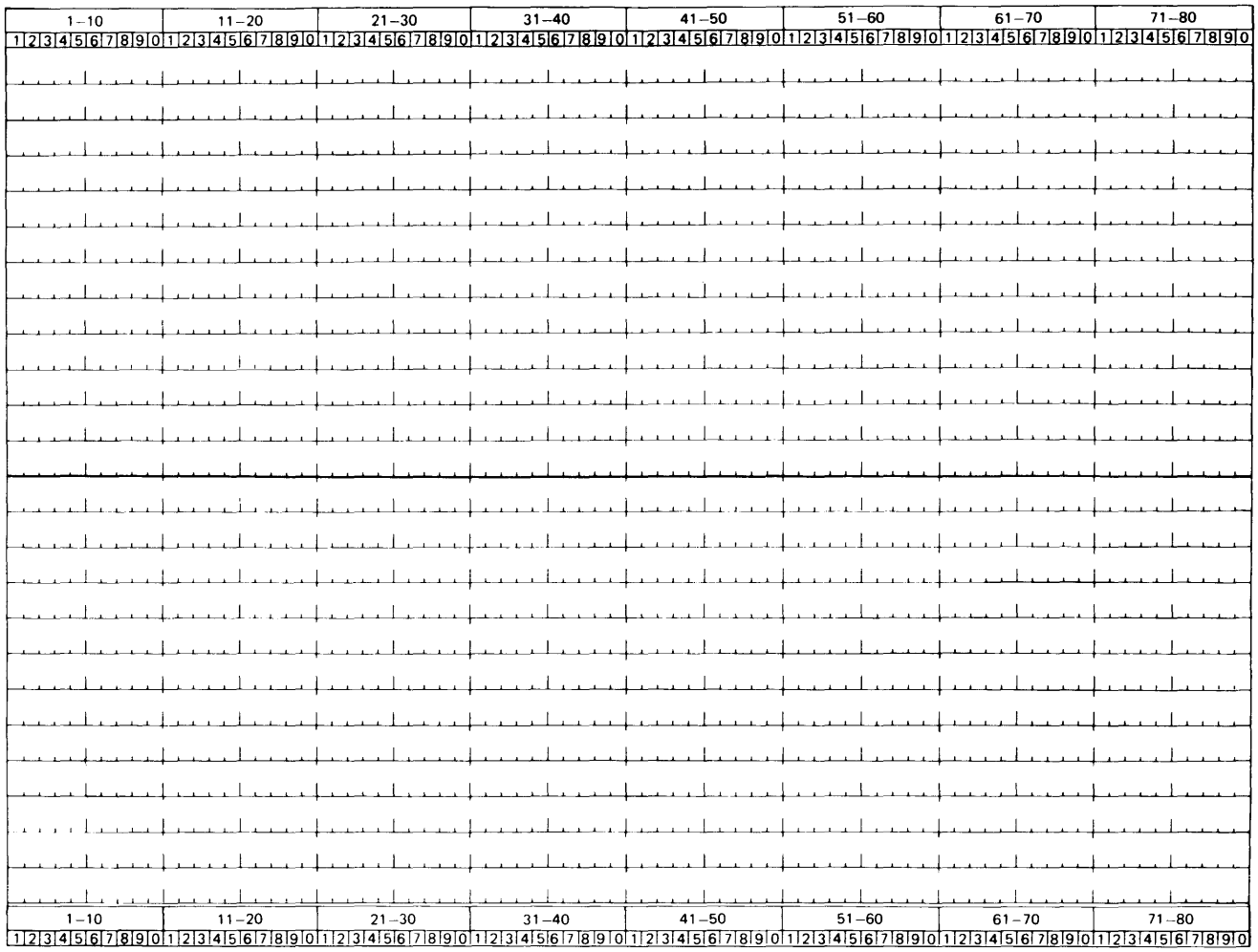
specification: The coding forms (A and Z) which have been designed to enable the coding of DE/RPG programs prior to their being entered into the system.

Appendix C. Blank Display Work Sheets and A and Z Specifications

Index

Display Screen Layout Sheet

COLUMN



IBM International Business Machines Corporation

IBM 5280 GENERAL UTILITY SPECIFICATIONS

Printed in U. S. A.

Job		Keying Instruction	Graphic								
Operator	Date		Key								

Description	Page of

Z

Sequence	Form Type	Name Type	Format ID (1-9, A0-Z9)	Job/Format/ Subroutine Name	Reserved	Repeat (1-9, N)	Mode (E/R)	Test Conditions: AND(A)	Test Conditions: Position to be Tested (*POSnnnn)	Test Conditions: Reserved	Test Conditions: Condition	Test Conditions: Character to Test for ('C')	Reserved	Next Format ID (0-9, A0-Z9)	Reserved	Options
1 2 3 4 5	6	7	8 9	10 11 12 13 14 15 16 17	18 19	20	21	22	23 24 25 26 27 28 29 30	31 32	33 34	35 36 37	38 39 40 41 42 43 44	45 46	47 48 49 50 51 52 53 54	55 56 57 58 59 60 61 62 63 64 65 66 67 68 69 70 71 72 73 74 75 76 77 78 79 80
01	Z							*	POS		EQ	' '				
02	Z							*	POS		EQ	' '				
03	Z							*	POS		EQ	' '				
04	Z							*	POS		EQ	' '				
05	Z							*	POS		EQ	' '				
06	Z							*	POS		EQ	' '				
07	Z							*	POS		EQ	' '				
08	Z							*	POS		EQ	' '				
09	Z							*	POS		EQ	' '				
10	Z							*	POS		EQ	' '				
11	Z							*	POS		EQ	' '				
12	Z							*	POS		EQ	' '				
13	Z							*	POS		EQ	' '				
14	Z							*	POS		EQ	' '				
15	Z							*	POS		EQ	' '				
	Z							*	POS		EQ	' '				
	Z							*	POS		EQ	' '				
	Z							*	POS		EQ	' '				
	Z							*	POS		EQ	' '				
	Z							*	POS		EQ	' '				

Options

Job Line

CFILE (data set)
DATE (*DMY/*YMD)
EDITC (cuptd)
ENTRATR ('attr...)
EXITATR ('attr...)
JOBOPT ([*NOPMT][*NOOPEN])
PRTFILE (data set)
SHARE (names)
SHARER (names)
STATUS (name)
TFILE (data set [delfreq])

'attr=BL,CS,HI,ND,RI,UL

Entry Lines

CLRL (number)
EOJ [(['job' dev] [*PASS])]
SLNO (line)
WRITE (name)

1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20 21 22 23 24 25 26 27 28 29 30 31 32 33 34 35 36 37 38 39 40 41 42 43 44 45 46 47 48 49 50 51 52 53 54 55 56 57 58 59 60 61 62 63 64 65 66 67 68 69 70 71 72 73 74 75 76 77 78 79 80

*Number of sheets per pad may vary slightly.

1-5 Identifies the source statement order.

6 Identifies the type of source statement.

7 Names the type of source statement:
*–User comment
J–Job specification
blank–Format specification

8-9 The identification associated with this format:
1 through 9–A single numeric character ID.
A0 through Z9–A two-character ID consisting of an alphabetic character followed by a numeric character.

10-17 The name used to:
- identify the job (J in column 7).
- identify the format or subroutine (blank in column 7).

These columns are not used if column 21 contains an R.

18-19 Reserved.

Note: Columns 20-54 are not used if column 7 contains a J.

20 Specifies the number of times the format is repeated before the next format is used:
1 through 9–Repeat the format for the specified number of times unless the SEL FMT or NEXT FMT key is pressed.
blank or N–Repeat the format until the SEL FMT or NEXT FMT key is pressed.

21 Specifies how the format is used:
E–(Entry) used to enter and display data.
R–(Review) used to select a format for scan, update, or verify of existing records.

22-37 Used for logical selection of a format. Multiple tests are allowed. In enter mode, the format selected is used to format the *next* record entered. In review mode, the format selected is used to display the *current* record.

22 In review mode (column 21 contains an R), an A specifies the *anding* of two characters in the data record to create a unique record identifier.

23-30 *POSnnnn identifies the position in the data record to be tested, where nnnn is a numeric value from 1 to 1024.

31-32 Reserved.

33-34 The characters EQ or blank when a character to test for is specified in position 35-37.

35-37 Specifies the character that controls format selection if it matches the character in the data record.

38-44 Reserved.

45-46 Specifies the identification of the format used for the entry or display of the next record. If columns 22-37 are specified, the format is selected when a match occurs. If columns 22-37 are not specified in enter mode (E in column 21), the format is selected when the repeat count (column 21) is met or the NEXT FMT key is pressed. If columns 22-37 are not specified in review mode (R in column 21), the format is selected if no previous match occurs.

47-54 Reserved.

55-80 Keywords that specify information used for jobs or formats:

JOB specifications (J in column 7):

CFILE (data set)–Includes the COPY function in the job. The parameter data set is the data set name from which records will be copied.

DATE(*DMY/*YMD)–The format of the date available in UDATE. The default is *MDY, where M = month, D = day, and Y = year.

EDITC(cuptd)–Five characters that define the editing control for output fields, where:
- cu is a two-character currency symbol (default = b$).
- p is the decimal point character (default = .).
- t is the thousand separator character (default = ,).
- d is the date separator character (default = /).

The system default for this option is b$.,/ if EDITC is not specified.

ENTRATR (attr...)–Specifies the attributes that are applied to all input/both fields only when the fields are being entered, where attr is:
BL (blink)
CS (column separators)
HI (high intensity)
ND (nondisplay)
RI (reverse image)
UL (underline)
A combination of attributes can also be used.

EXITATR (attr...)–Specifies the attributes that are applied to all input/both fields after the fields have been entered. See the ENTRATR for a description of the attr parameter.

JOBOPT([*NOPMT] [*NOOPEN])–At least one of the parameters must be specified. Where:
- *NOPMT specifies to bypass the prompts for data set information at the beginning of the job.
- *NOOPEN specifies to bypass the automatic opening of all files except the transaction file specified by the TFILE keyword.

PRTFILE (data set)–Includes the PRINT function in the job. The parameter data set is the data set name to be assigned to the printer.

SHARE(names)–Allows other programs to read or write records in the data set specified by the names parameter while this program is executing.

SHARER(names)–Allows other programs to *read* records in the data set specified by the names parameter while this program is executing.

STATUS(name)–Establishes a variable that can be used to check the status of an I/O device after an I/O operation. The parameter name is the name assigned to the variable.

TFILE(data set [delfreq])–Specifies the data set where records will be written after a format is completed, where:
- data set is the name of the data set that receives the transaction records.
- delfreq specifies how often deleted records are automatically inserted in the transaction data set.

Format Specifications (blank in column 7):

CLRL (number)–Specifies the number of display lines cleared, starting from the first line of the display, when a new record is to be entered. If *NO is specified, none of the display lines are cleared.

EOJ [('job' dev] [*PASS])]–Causes the end of the job upon completion of the format. The optional parameters are:
job–name of the next job to execute.
dev–the device address where the next job is located.
*PASS–suppress job production statistics.

SLNO (line)–Specifies the uppermost display line that can be used. All display line references are based on the specified line as line one.

WRITE (name)–Specifies that the current data is written to the data set in the record format specified by *name*. If *NO is specified, the current data is not written to the data set.

Continuation can be specified by a + or - as the last character on the line, where:
+ specifies to continue with the first nonblank character in positions 55-80 on the next line (ignore leading blanks).
- specifies to continue from position 55 on the next line (including leading blanks).

IBM International Business Machines Corporation

IBM 5280 DATA DESCRIPTION SPECIFICATIONS

Printed in U.S.A.

Job No.	Dataset	Keying Instruction	Graphic							
Operator	Date		Key							

Source Document	Page of

A

Sequence	Form Type	Comment (*)	Reserved	Indicator (for CHECK (BY, BV) or ERROR)	Reserved	Name Type (F/R/K/T)	Reserved	Dataset/Record/ Field/Table Name	Reserved	Length	Data Type	Reserved	Decimal Positions (0-9)	Usage (I/O/B/W)	Location: Line	Location: Pos	Editing
1 2 3 4 5	6	7	8	9 10	11 12 13 14 15 16	17	18	19 20 21 22 23 24 25 26	27 28 29	30 31 32 33 34	35	36	37	38	39 40 41	42 43 44	45 46 47 48 49 50 51 52 53 54 55 56 57 58 59 60 61 62 63 64 65 66 67 68 69 70 71 72 73 74 75 76 77 78 79 80
01	A																
02	A																
03	A																
04	A																
05	A																
06	A																
07	A																
08	A																
09	A																
10	A																
11	A																
12	A																
13	A																
14	A																
15	A																
	A																
	A																
	A																
	A																
	A																

Checks=CHECK (code . . .)

Auto Dup	=AD	Mandatory Entry	=ME
Auto Skip	=AS	Mandatory Fill	=MF
Blank Check	=BC	Rt Adj–Blank Fill	=RB
Bypass	=BY	Right to Left	=RL
Bypass on Verify	=BV	Rt Adj–Zero Fill	=RZ
Data Required	=DR	Self-Check	=nxx
Dup Disable	=DD	n M/G (Check/Gen)	
Field Exit Required	=FE	xx Modulus	
Lower Case	=LC		

Functions

ADD (name)
AUXDUP (name)
AUXST (name)
COMP ([1]test fld1[3]@ . . . fldn 'literal' [indicator])
DSPATR ([2]attr . . .)
EDTCDE (code 'float')
ERROR (code ['message'])
EXSR (subroutine)
INSERT (fld1[3]@ . . . fldn 'literal')
LOOK (table [index])
PMT (prompt)
RANGE (low high)
RANGET (table [index])
RESET ([*TOTn] [name])
SEQ ([1]test)
SETOF (ind)
SETON (ind)
SHIFT ([4]shift)
SUB (name)
SUBST (table1 table2 [index])
TADD ([*TOTn] [name])
TSUB ([*TOTn] [name])
XCHK (table index1 index2)
'literal'

[1]test=EQ,GE,GT,LE,LT,NE
[2]attr=BL,CA,CS,HI,ND,RI,UL
[3]@=+,–,*,/
[4]shift=A,D,H,N,V,W,X,Y

1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20 21 22 23 24 25 26 27 28 29 30 31 32 33 34 35 36 37 38 39 40 41 42 43 44 45 46 47 48 49 50 51 52 53 54 55 56 57 58 59 60 61 62 63 64 65 66 67 68 69 70 71 72 73 74 75 76 77 78 79 80

*Number of sheets per pad may vary slightly.

1-5 Identifies the source statement order.

6 Identifies the type of source statement.

7 An * indicates a user comment.

8 Reserved.

9-10 Specifies the indicator that is used to control field bypassing or displaying user error codes.

11-16 Reserved.

17 Defines the type of statement: F = data set, R = record, K = key field, T = table, blank = field.

18 Reserved.

19-26 Specifies the name for: data set (max 8 characters), record (max 8 characters), field (max 6 characters), or table (max 6 characters).

27-29 Reserved.

30-34 Specifies the length:
data set = maximum record length is required.
record = number of characters (1-8192)
field = number of characters (1-256 for alphameric or 1-16 for numeric).

35 Defines the data type for the field:

A = alpha	X = alpha only
N = numeric	Y = numeric only
S = signed numeric	D = digits only
W = Right half shift	V = Right half only
H = hexadecimal	C = Use SHIFT keyword
	b = alpha or numeric depending on the field type

36 Reserved.

37 Specifies the number of decimal positions (0-9).

38 Specifies how the data in a field on the display screen is processed: I = input, O = output, B = both, W = workspace.

39-44 Specifies the location of the field within a record or on the display screen.

45-80 Specifies parameters for data sets, files, records, tables, and fields:

Data sets and files (F in column 17)

BLKING ([*DBL] [*FMTU or *FMTS])–specifies blocking characteristics for data sets:
*DBL specifies to use two physical buffers
*FMTU specifies that the records are unblocked (Basic or H data exchange)
*FMTS specifies that the records are blocked and spanned (I data exchange)

DEVICE (dev-type address)–physical device type for the data set:
dev-type is COMM (communications), CRT (keyboard/display), DISK (diskette), MREAD (magnetic stripe reader), PRINTER (printer).
address is the 2-character logical ID or the 4-character device address (X'xxxx' where xxxx is the physical address).

DSPSIZ (lines 80)–specifies display size: lines = 6, 12, or 24.

FORM (length [overflow-line overflow-ind])–Specifies the printer page size; length specifies the lines available on the page, overlow-line specifies the line that sets the overflow indicator on, and overflow-ind specifies the indicator that is set on.

INDEX ([storage] [data set])–At least one parameter must be specified. Specifies the storage reserved for the sparse index and the index data set name: storage specifies the space required for the index, data set specifies the name of the index data set.

LABEL (name of data set)–diskette data set name.

LOGON ('message' or name)–Specifies the log on information when required for communications. The parameter can be either a message enclosed in single quotes or a variable name.

NUMENT (number)–number of records in a data set when used for dynamic allocation of the data set or the number of entries in a table.

Records (R in column 17)

DSPATR (attr...)–Specifies the display attributes that apply to all the fields in the record.

MARK (*POSnnnn)–Specifies the position in a data record where an E is placed if the Field Mark key is pressed.

VMARK (*POSnnnn)–Specifies the position in a data record where a V is placed after the record is verified.

RECID (*POSnnnn 'c')–Specifies the position that identifies the single character record type 'c' from a data set with more than one record type (nnnn is 1 to 8192).

SPACEA (n)–Causes the printer to space n lines after the record is printed.

SPACEB (n)–Causes the printer to space n lines before the record is printed.

SKIPA (n)–Causes the printer to skip to line n after the record is printed.

SKIPB (n)–Causes the printer to skip to line n before the record is printed.

Field (Blank in column 17)

ADD (name)–Adds the data in the current field to the named field with decimal alignment.

AUXDUP (name)–Duplicates data from the named field if the Dup key is pressed or the Auto Dup/Skip switch is on.

AUXST (name)–Stores the current field in the named field if the Auto Dup/Skip switch is on.

CHECK (parameter)–Specifies the keyboard edits to be applied to the field.

COMP (test fld1@...fldn 'literal' [indicator])–Compares the current field with a named field, the specified expression, or a literal and optionally turns on an indicator if the compare is true.

DSPATR (attr...)–Controls the display attributes for each field.

EDTCDE (code ['float'])–Specifies the editing that is to be applied to data in numeric fields, where:
code is a single character that controls the use of editing characters specified by the EDITC keyword.
float can be either:
*, which places asterisks in the character positions to the left of the first digit
cu, which floats the two-character currency symbol used on EDITC.

ERROR (code ['message'])–Locks the keyboard, displays an error code, and optionally displays an error message (when the Help key is pressed) if the specified indicator is turned on.

EXSR (subroutine)–Branches to the named calculation subroutine.

INSERT (fld1@...fldn 'literal')–Inserts the named field, expression or literal into the current field.

LOOK (table [index])–Compares the current field for a match in a table, and optionally places the index value of the table entry in index.

PMT (prompt)–Displays the prompt message when the current field is entered.

RANGE (low high)–Specifies the low and high limits for data that can be entered into the current field.

RANGET (table [index])–Compares the current field for a match in a table of low and high limits, and optionally places the index value of the table entry in index.

RESET ([*TOTn][name])–Only one parameter is allowed. Sets the named counter to 0.

SEQ (test)–Sequence checks the data in the current field against the data from the previous sequence check using the specified test.

SETOF (ind)–Turns the specified indicator off.

SETON (ind)–Turns the specified indicator on.

SHIFT (shift)–Specifies the shift and character set for each character in a field when C is specified for data type.

SUB (name)–Subtracts the data in the current field from the named field with decimal alignment.

SUBST (table1 table2 [index])–Compares the current field for a match in table1. If there is a match, replaces the current field with data from the corresponding entry in table2. Optionally places the index value of the table entry in index.

TADD ([*TOTn][name])–Only one parameter is allowed. Adds the current field to the named counter.

TSUB ([*TOTn][name])–Only one parameter is allowed. Subtracts the current field from the named counter.

XCHK (table index1 index2)–Compares the indexes to see if they match an entry in a named table of index pairs.

Continuation–Specifies to continue on the next line:
+ specifies to continue with the first nonblank character in position 45-80 on the next line (ignore leading blanks).
- specifies to continue from position 45 on the next line (leading blanks are included).

READER'S COMMENT FORM

Please use this form only to identify publication errors or request changes to publications. Technical questions about IBM systems, changes in IBM programming support, requests for additional publications, etc, should be directed to your IBM representative or to the IBM branch office nearest your location.

Error in publication (typographical, illustration, and so on). **No reply.**

Page Number *Error*

Inaccurate or misleading information in this publication. Please tell us about it by using this postage-paid form. We will correct or clarify the publication, or tell you why a change is not being made, provided you include your name and address.

Page Number *Comment*

IBM may use and distribute any of the information you supply in any way it believes appropriate without incurring any obligation whatever. You may, of course, continue to use the information you supply.

Name _____

Address _____

_____

● No postage necessary if mailed in the U.S.A.

IBM 5280
Distributed Data
System
Introduction to DE/RPG

SC21-7803-1

Cut Along Line

Fold and tape | Please do not staple | Fold and tape

NO POSTAGE
NECESSARY IF
MAILED IN THE
UNITED STATES

BUSINESS REPLY MAIL

FIRST CLASS PERMIT NO. 40 ARMONK, N. Y.

POSTAGE WILL BE PAID BY ADDRESSEE:

IBM CORPORATION
Product Information Development, Dept. 997
11400 Burnet Road
Austin, Texas 78758

Fold and tape | Please do not staple | Fold and tape

IBM

International Business Machines Corporation

General Systems Division
4111 Northside Parkway N.W.
P.O. Box 2150
Atlanta, Georgia 30301
(U.S.A. only)

General Business Group/International
44 South Broadway
White Plains, New York 10601
U.S.A.
(International)

International Business Machines Corporation

General Systems Division
4111 Northside Parkway N.W.
P.O. Box 2150
Atlanta, Georgia 30301
(U.S.A. only)

General Business Group/International
44 South Broadway
White Plains, New York 10601
U.S.A.
(International)

IBM 5280 Distributed Data System Introduction to DE/RPG (File No. 5280-28) Printed in U.S.A. SC21-7803-1